Assembling your own Enterprise system

/ R.J.H. ROTTING

/ UNIVERSITY OF TWENTE
/ MASTER THESIS

NOVULO

UNIVERSITY OF TWENTE.

08-07-2015

UNIVERSITY OF TWENTE.

Assembling your own Enterprise system:

Tools enabling small- and medium enterprises in designing and implementing their own web-based enterprise system

Master thesis

Written by R.J.H. Rotting

to receive the degree Master of Business Administration

Information Management track

at the University of Twente

Faculty of Management & Governance

Department of Industrial Engineering and Business Information Systems (IEBIS)

Supervisor Dr. M.A. Iacob

Second supervisor Dr. Ir. A.A.M. Spil

Supervisors Novulo F. Wille

W. Badenhop

Abstract:

Implementations in a SAAS environment require fewer activities to complete. In this thesis the possibility of implementing your own web based enterprise system as an SME has been examined. To this end multiple models have been created and validated; (1) a model implementation process, (2) an integrative model combining theory from a literature research and the implementation process, and (3) a prototype tool using Novulo software to validate the models. Being able to successfully implement your own system cuts consultancy costs, while providing all the benefits of having an enterprise system.

The results show that with tool supports the implementing activities regarding the training, general introduction, and (simple) data migration. The configuration of the software is still too complex and requires support from a consultant to successfully complete, albeit a significantly lower amount. Contributions to the literature include the implementation model, the integrative model and the research results.

Keywords: ES implementation – SAAS – Web based implementation – Consultancy effort – Implementation process – Prototype

Date: *30-6-2015*

PREFACE

In this preface I would like to express my thanks to the people that cooperated to the research included in this thesis and give a short overview of the work that has been put into writing this thesis. My interest in the successful implementation of enterprise systems (ES) comes from my motivation to use the information (data) within a company as efficient as possible. I believe that companies can learn a lot from this information and have much to improve in this regard. An enterprise system is the first step towards this goal, providing the opportunity to store information efficiently and orderly way. The use of ES in SMEs is still lacking, making this the perfect market for this research.

With my choice of the company I had already worked for 1,5 years as the case company has had his up- and downsides. This made the amount of research into the company's vision, standards and ethics easier, but it also proved to be something that would lengthen the time span needed to complete the overall research. While present at the company, the workload that had already been there before the start of the writing of the thesis continued. This meant a choice between working on the companies projects and the writing of the thesis. To end this dilemma, establishing a final deadline proved to be the key. Thankfully the support I got from Novulo and the employees helped a lot.

I would to thank the people that helped me during the writing of this thesis. They are the ones that made it possible for me to complete this research in the end and I thank them greatly for it. Firstly, I would like to thank my supervisors from the university, Maria and Ton, and my supervisors from Novulo, Frank and Willem, for their help and support during the process of writing this thesis. I would also like to thank the employees at Novulo for their support during the thesis and the interviews that made designing the implementation process possible. Special thanks to Deni for his contributions to the tool and Helena for proof reading. The company owners willing to spend their time on the workshop were also essential during the validation of the models. I would therefore like to thank Samantha, Leon, Henri, Wim, Michael, Tom, and Niels for their time and effort.

I hope you enjoy reading this thesis and that it contributes towards your insight in web-based enterprise systems (implementation).

Remy Rotting

Enschede, June 30, 2015

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

FIGURES

TABLES

MANAGEMENT SUMMARY

An Enterprise system contributes to technical areas such as standardization, transparency, and globalization and assists enterprises in automating and integrating corporate cross-functions. Pereira indicated that a well-managed ES could be determinant of strategic competitive advantage and a successful ES implementation is proven to increase the chances of long term survival of an organization. The thesis follows the definition portrayed by several ES and ERP studies, defining an enterprise system as: "*a comprehensive, packaged software solution seeking to integrate the complete range of a business's processes and functions in order to present a holistic view of the business from a single information and IT architecture*".

With the growth of software being distributed online (software as a service, or web based enterprise systems as examples), the need (and possibility) to decrease the effort is also growing. With the Software as a Service (SAAS) model a customer contracts the use of an enterprise system, such as ERP or CRM, hosted by a third party, rather than buying a software license and installing the application on its own machines.

The main goal of the research was to find what factors influence the work (or activities) the implementing company can do themselves during the implementation stage of an enterprise system, and how these factors can be influenced by a tool. The goal is to examine if it is possible for an SME to implement their own Enterprise system alone, given a supportive tool to guide them through the process.

To summarize, the following contributions to literature can be found in this study:

1. A model describing the implementation process of a web based Enterprise system
2. An integrative model combining a literature research and the implementation process into factors influencing the amount of activities
3. Validation to these models by testing a prototype based on the above models

The implementation process

The model used in this study is the model by Markus and Tanis. The model consists of four phases, characterized by key players, typical activities, characteristic problems, appropriate performance metrics, and range of possible outcomes. The phases are: project chartering, the project, shakedown, and the onward and upward phase. The project phase describes the implementation phase and is the main focus in this research.

The implementation model by Markus and Tanis has been adapted to better suit the implementation process during a web based implementation. Interviews have been held with consultants at Novulo to ensure the validity of the activities still remaining in this kind of implementation. For each of the implementation phases, the inputs, outputs, activities, and techniques have been identified. The process described the entire implementation, including the possibility of new iteration cycles when changes are desired by the implementing company.

The integrative model

The literature research and the implementation process are combined into an integrative model. The **business architecture** (or organizational context) includes the companies' technical, organizational, and environmental characteristics. These characteristics explain the knowledge, resources and restraints within the implementing company.

The **technical architecture** of the implemented system has a large impact on the amount of activities that are needed to implement the system. Implementing a web based (SAAS) enterprise system, instead of a traditional on-site implementation, eliminates many of the activities, creating the possibility to implement your own system.

The **system architecture** has an impact on the amount of training that is required to work with the enterprise system. When the system is easy to use and the processes fit with the current business processes, the amount of time to transfer to the new system is decreased.

The factors of the technical and the system architecture influence the secondary factors: the number of implementing activities and the amount of training required. These secondary factors can be influenced by the tool.

The **implementing activities** remaining in a SAAS context have been discussed before. The **training requirement** is determined by the ease of use of the software, the amount of BPR required and knowledge within the company.

With the ease of use of the system and the fewer remaining activities, it could be possible to implement a system without the use of consultants. The **tool** must ensure that the abilities of the implementing company are sufficient to successfully implement their own system.

The prototype

The main goal of the prototype is to provide all the support a lead user needs to implement a web based enterprise system in their own company. The prototype is based on the model described in the literature review and will test the possibility of implementation done entirely by the implementing organization. This means that all implementing activities named by Markus and Tanis, if relevant for the implementation, will have to be supported by the prototype tool. The implementation process can be divided into several functional parts.

These parts are the various implementation activities grouped into four subjects that are held together by the implementation process itself. These subjects are described in detail below:

- **Implementation process:** The process that encompasses the entire implementation, from start to finish. Since knowledge about this process is lacking in an SME, the tool should guide the implementing company through this process, with clear directions and milestones.
- **General:** Instructions in the general use of the application. Subjects of the integrative model included in this segment are the general system design and use of the application, providing the basic knowledge that is essential for the use of the system.

- **Training:** This subject is based on the training activities that are required during an implementation project. It discusses the various processes a user has to do in their daily routines, like customer, sales, or purchase registration.
- **Configuration:** Configuration is everything that is related to the set-up of the system. Configuring these settings and checking them usually requires knowledge about the enterprise system.
- **Data:** The data subject handles the conversion and migration of the legacy data, if any. This choice can be made on several levels, depending on the desired amount of data that needs to be converted.

The analysis

To analyze the results coming from the workshops, a survey has been implemented into the workshop. The selected statements covered a variety of aspects of system usability, such as the need for support, training, and complexity, and thus have a high level of face validity for measuring usability of a system.

Overall implementation success: The overall implementation score can be seen as an affirmation of the usefulness of the tool.

General: The scorings in this part showed the difference in the knowledge requirement in different parts of the application; the introduction in the general use was scored highly, but the use and configuration of accounts, and especially rights profiles, were scored significantly lower.

Training: The training instruction showing the business process within the application had the highest usability scorings. They were found to be helpful by all participants and showed that the processes within Novulo can be explained by a tool in such a way that the implementing company can complete the activities linked to these processes successfully.

Configuration: All participants found completing the configuration the hardest part of the implementation. The validation and set up of the settings was found to be complex and participants could not link the settings to the context with the amount of knowledge they had gained through the tool.

Migration: With the support of the tool and the Excel sheets provided by Novulo, the participants were able to successfully import some of their customers into the system.

Conclusion

The results show that with the tool supports the implementing activities regarding the training, general introduction, and (simple) data migration. The configuration of the software is still too complex and requires support to successfully complete, albeit a significantly lower amount. Contributions to the literature include the implementation model, the integrative model and the research results. Suggestions for future research are further validation for the model and testing the model in different contexts.

1 INTRODUCTION

An Enterprise system contributes to technical areas such as standardization, transparency, and globalization [1, 2] and assists enterprises in automating and integrating corporate cross-functions. Pereira [3] indicated that a well-managed ES could be determinant of strategic competitive advantage and a successful ES implementation is proven to increase the chances of long term survival of an organization [4]. Tangible benefits might include better visibility of future requirements, improved material control, reduced costs, increased productivity, increased on-time deliveries, improved customer service, and the elimination of redundant and contradictory data bases. Intangible benefits might include improved communications, substantially reduced chaos and confusion, and higher morale [5]. These benefits have been thoroughly investigated in literature in the last two decades, but with the implementation of an Enterprise system also come considerable risks, as numerous cases have shown in the past [6] [7] [8]. Even with the development of midrange and less complex systems specifically for SMEs, implementation remains a challenge [9].

1.1 PROBLEM STATEMENT

99 out of 100 companies in the EU are SMEs [10], and with the market for larger, global companies getting more and more saturated, these companies are the new targets for business software developers/suppliers, as shown by the increase in SME specific systems and the amount of adoption by SMEs [11]. However, the SME market requires a different, more cost-focused, approach then global companies. The differences are obvious; the amount of resources (human and financial) available to implement a system, the limited IS knowledge, and lack of expertise in IT [12] [13]. Making it easier to implement a system, and ultimately being able to implement your own system, could improve the willingness to adopt or switch to a certain enterprise system.

With the growth of software being distributed online, with Software as a Service, web based enterprise systems as examples, the need (and possibility) to decrease the effort is also growing. This online distribution of software provides multiple benefits for the implementing company, like lowering the total cost of ownership and improved insight into these costs. However, the adoption of online Enterprise systems has been lacking compared to other areas where the SAAS concept has been utilized [14]. Improving the ability of the implementing company to complete the implementation process themselves could boost the adoption of Enterprise systems delivered through the SAAS concept.

The implementation effort is usually split between activities that can be done by the implementing company and activities that require the use of external consultants. When focusing on online distribution this need for consultants will be a hindrance. This amount is to be decreased by following the recommendations coming from the literature on what factors influence the ability of users to implement their own system in a web based

environment. Literature on implementations in this environment is still lacking in several areas, e.g. the different (shorter) life-cycle compared to standard on premise enterprise systems.

Software as apps

The concept of 'apps' has been a hot topic the years following the introduction of smartphones: the iPhone in 2007, its App Store, and the introduction of Android. 'App' is the abbreviation of 'application', meaning a computer program that is written and designed for a specific need or purpose [15]. The use of apps, or 'modules' in Enterprise systems has been mentioned in literature since 2000 [16], stating that it will be the next step in being able to customize your own system. The use of apps in the context of Enterprise systems can for example be found in separating the various sectors (HR – Production – Customer Relations – Finance), since each have their own separate functions.

The introduction of an 'app store' that's comparable to the well-known stores from the Apple and Android operating systems could provide the familiarity needed for business users to pick the functionality they want (and need), thus enabling them to design their own Enterprise systems. This concept of 'app stores' for enterprise systems will not be researched in this study, but provides further insight into the need for decreasing the implementation effort; if you're able to design your own system, why not implement it yourself too?

1.2 SCIENTIFIC RELEVANCE

The scientific aim of the study will be to further knowledge about the implementations of enterprise systems in SME's, especially in the way effort can be reduced during these implementations. With more and more focus on online distribution, reducing the amount of effort, making it possible for business users to implement their own system, is an important factor in increasing the success of these formulas.

The concept of effort during implementation is one that has not been researched thoroughly yet in literature, which is most likely due to the fact that consulting activities comprise a significant amount of income to software developers/suppliers. The focus of researchers has been on how to achieve implementation success itself, not how to best achieve it from the perspective of effort needed. The research into Dutch SMEs and implementations of Enterprise systems is also lacking, giving this study further value.

The experiment that will be conducted at the end of this study will provide practical knowledge and insight on the usefulness of the theoretic framework and model developed in this study. The translation from theory to practice is an important step and the experiment can further the knowledge on how this translation can be done. The experiment will show if it is possible for a company to design and implement their own system, providing the answer to the main research question from a more practical point of view.

The theoretical model, together with the outcomes and implications from the experiment, will provide the evidence needed to answer the problem that is at the core of this study: Is it possible for a business user to implement their own enterprise system, when sufficient tools are provided according to the factors influencing effort during the implementation of an enterprise system? This question, be it in a somewhat different form, and the sub questions that follow from it, will be discussed next.

1.3 RESEARCH QUESTION

As mentioned before, the main goal of the research is to find what factors influence the work (or activities) the implementing company can do themselves during the implementation stage of an enterprise system, and how these factors can be influenced by a tool. **The goal is to examine if it is possible for an SME to implement their own Enterprise system alone, given a supportive tool to guide them through the process.** To find an answer to this problem, the following research question will be adopted in this study:

> What factors (methods and techniques) enable an SME to design (based on existing functional components) and implement their own enterprise system?

The research question will be split into smaller questions. These questions will be used in the literature research and an integrative model will be designed and tested during an experiment. The sub questions that will be discussed in the study are the following:

1. What are web based enterprise systems (ES)?
2. What is the reference implementation process of a web based ES in (Dutch) SMEs?
3. How can you define the 'ideal' implementation process, which can be built into a tool?
4. What factors influence the amount of effort needed to successfully implement a web based ES in an SME?
5. How can you design this implementation tool as intuitive as possible?

1.4 HYPOTHESES

Multiple hypotheses will be tested to see if the model (and the tool) is valid. The hypotheses are listed below, starting with the main hypothesis.

H_0: It is possible for the implementing company to complete all the activities needed to implement a web based enterprise system

1. H_1: It is possible to develop a project planning for the implementation of a web based enterprise system using only a tool provided by the vendor
2. H_2: It is possible for the implementing company to instruct the other users in their daily processes using only the training included in a tool provided by the vendor

3. H_3: It is possible for the implementing company to fully configure the enterprise system using only a tool provided by the vendor
4. H_4: It is possible for the implementing company to complete the conversion and migration of legacy data using only a tool provided by the vendor

The hypotheses will be tested during the experiment, described in chapter 6. In this chapter the research model and the model implementation process will be combined and analyzed to provide proof for these models.

1.5 THE DESIGN SCIENCE APPROACH

The design process of the prototype will follow the iteration steps as provided by Peffers [17], which is described below. This process is also the core of the research path in this thesis.

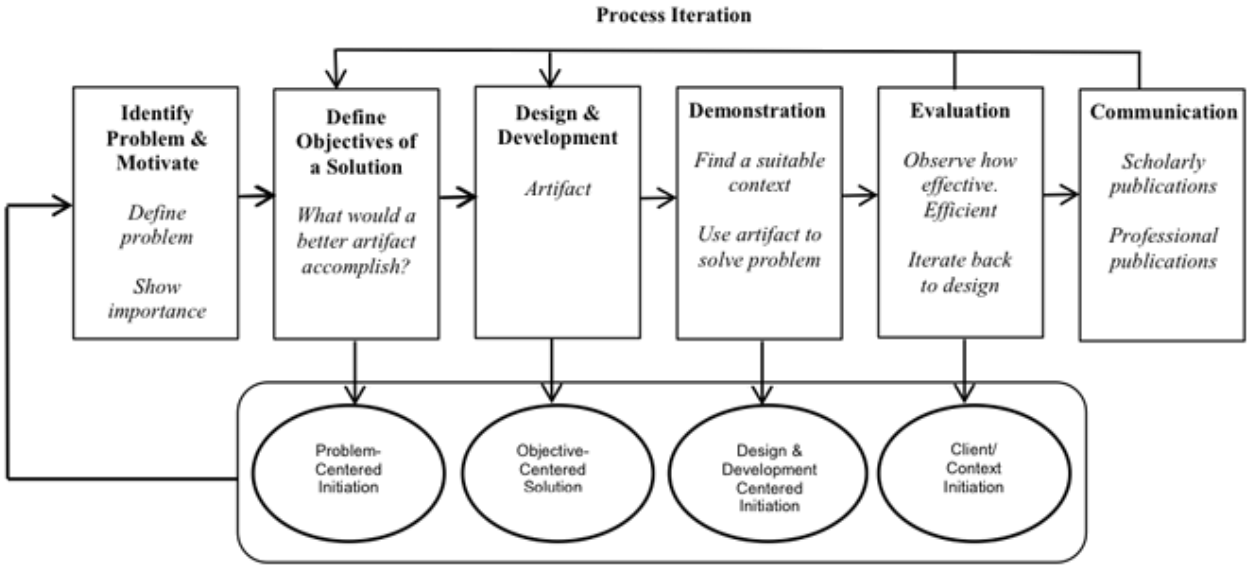


Figure 1: The design science approach by Peffers

Identify Problem & Motivate

The definition of the problem and its motivation can be found in chapter 1, with the research problem and statement. These can be found in chapter 1.1, and chapter 1.3.

Define Objectives of a Solution

The solution is offered in the form of an integrative model, which was created from the literature research. The various factors found during the research have been integrated, to form a model that provides insight into the activities that remain in a web based enterprise system implementation. The model is found in chapter 3.3.

Design & Development

The design and development of the solution (the artifact, or the prototype) is explained in this chapter. This is done according to the practice rules provided by Hevner et al. [18]:

1. The research must produce an artifact created to address a problem.
2. The artifact should be relevant to the solution of an unresolved and important business problem.
3. The utility, quality and efficacy must be evaluated.
4. The research should represent a verifiable contribution.
5. Rigor must be applied in both development and evaluation of the artifact.
6. The development of the artifact should be a search process that draws from existing theories and knowledge to come up with a solution to a defined problem.
7. The research must be effectively communicated to appropriate audiences.

All of these guidelines are discussed in this research in one way or another. The artifact produced is the prototype tool, it is relevant for the solution of the research problem defined in this study. The tool is evaluated by surveys and results from an experiment. It presents a contribution by trying to prove the implementation of a web based enterprise system no longer has to involve consultants. The rigor is provided by using a research methodology that has been proven as valid. The development of the artifact was started after an integrative model was made that draws from the current body of knowledge on business software implementation.

Demonstration

The demonstration of the artifact is done in chapter 6.1. In this chapter the experiment is discussed. The experiment demonstrates the use of the artifact (the prototype) to solve the problem (not being able to implement a web based enterprise system). Resources required for the demonstration include effective knowledge of how to use the artifact to solve the problem [17], which will be explained in chapter 6.2.

Evaluation

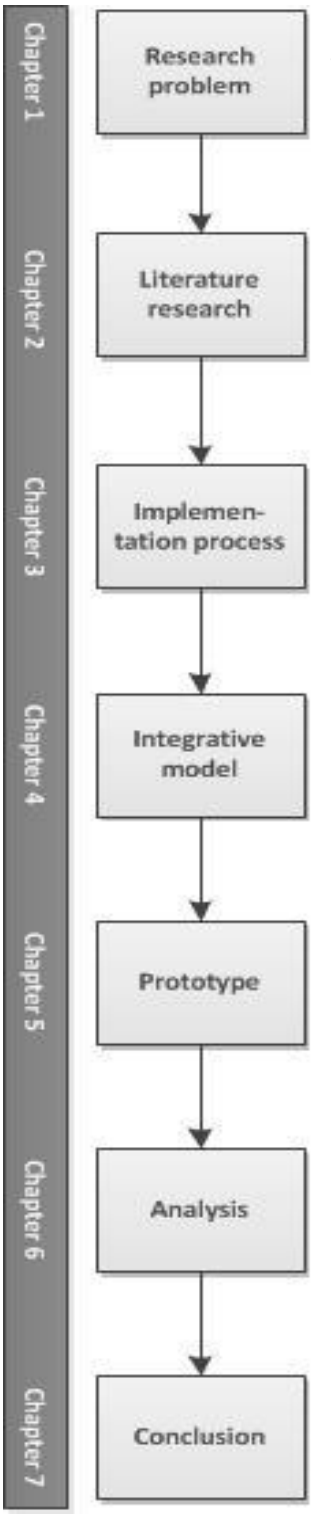
The evaluation of the demonstration will be done in chapter 6.3. The evaluation will require the results of a survey conducted after the demonstration. The survey design is also discussed in chapter 6.3 and describes and provides arguments for the measured variables. The conclusion drawn from the results and their evaluation can be found in chapter 7.

Communication

The communication of the problem, its importance, the artifact, its utility and novelty, rigor and effectiveness is the last step in the research design process. To provide this communication, a management summary is provided on page 7 and the conclusion discusses the recommendations and suggestions for future research.

1.6 RESEARCH PATH

The research path is explained below; this path will also be the structure of this paper.



The research problem has been explained in this chapter;

literature research will be done in chapter 2;

this literature research will be complemented with **interviews**, to find the reference **implementation process** in chapter 3,

to put this process to the test, an **integrative model** is created in chapter 4, combining the literature research and the implementation process.

From this integrative model **a prototype** is created in chapter 5,

followed by an experiment (analysis) that tests the prototype in a practical situation in chapter 6.

Final conclusions, recommendations and further research are discussed in chapter 7

Figure 2: The research path

2 LITERATURE REVIEW

This chapter of the study focuses on the existing research related to the topics discussed in the research question. As stated, a substantial amount of research has already been done in the field of Enterprise systems implementation. This research will not discuss all literature in that field, but focus on providing clarification of the concepts used in the study and strive to provide an overview of the factors influencing *a web-based Enterprise System implementation in an SME*. The main goal of the literature research in this paper is to find theories that provide answers to the research questions and combine them into a framework of recommendations (sub questions 1 - 3).

Literature research has been done by using the following sources:

- Search on Scopus, Elsevier, and Google Scholar:
  - Software implementation; ERP implementation; Implementation in SME; ERP as a service
  - (enterprise) Software as a service implementation yielded no results
- MIS Quarterly & Journal of MIS:
  - Last 5 years on ERP implementation and SME context
- References of articles
  - S.-I. Chang, S.-Y. Hung, D. C. Yen and P.-J. Lee, "Critical Factors of ERP Adoption for Small- and Medium- Sized Enterprises: An Empirical Study," *Journal of Global Information Management,* vol. 18, no. 3, pp. 82-106, 2010.
  - Huang, "A Compilation Research of ERP Implementation Critical Success Factors," *Issues in Information Systems,* vol. XI, no. 1, pp. 507-512, 2010.

In **chapter 2.1**, the ***background of enterprise systems*** will be discussed, by describing the development in these systems over the last decades. This will provide insight into the growth of ES, showing the need for constant innovation in this field. After the introduction into enterprise systems, the impact of the ***SME context*** is discussed in **chapter 2.1.1**. This context has significant influence on the implementing company's ability to implement their own system.

After that the ***context of web based enterprise systems*** will be clarified further (**chapter 2.1.2**). What is a web based Enterprise system and what are the differences (if any) with other information systems, like ERP or CRM and on-site implementations? What kind of profits can a web based ES provide for the organization (compared to on-site systems)? Why do organizations choose to adopt a system and what are the implementation risks?

Chapter 2.2 discusses ***the implementation process*** itself, the definition of implementation used in this study will be further clarified, as will the stakeholders and the implementation types. The various ***stages during an implementation*** of an enterprise system are described in **chapter 2.2.1**.

There are multiple views on these stages and the factors indicating the success or failure of each stage. The different theories are discussed and a model will be adopted for this study. The activities that are at the core of an implementation are described in detail, since these activities will have to be done by the implementing company if the model is to be validated.

In **chapter 2.2.2**, the SAAS (or ERP-as-a-Service) specific factors are explained and the ***effort*** and ***participation*** during the implementation stage is discussed. Since the topic of web-based implementations is relatively new (subjects have only been studied since 2008), the remaining gaps in the literature will be filled by the information provided by interviews with employees/consultants of the software vendor (Novulo) in chapter 3.

The last chapter (**2.3**) describes methods that are of use for the ***design and practical usage of a prototype***. Topics like the step-by-step design and the use of gamification methods are mentioned. This literature research ends with a concluding statement about the factors influencing the implementation effort and the ability of an organization to implement its own system.

2.1 WEB-BASED ENTERPRISE SYSTEMS

Most research on Enterprise systems, and the implementation of Enterprise systems, has been done on large, global organizations, with on-site implementations of custom made software systems. These organizations were the first adopters of Enterprise systems, so the literature analyzed these organizations first. With the market for larger, global companies getting more and more saturated, the new targets for business software developers and suppliers are the Small and Medium Enterprises, as shown by the increase in SME specific systems and the amount of adoption by SMEs [11].

Since there are various definitions of Enterprise Systems (ES) in the literature, the definition used in this study will first be discussed here. This study, in line with Markus and Tanis [19], perceives enterprise systems as a more generic term for an information system used in an organization.

The thesis follows the definition portrayed by several ES and ERP studies, defining an enterprise system as: "***a comprehensive, packaged software solution seeking to integrate the complete range of a business's processes and functions in order to present a holistic view of the business from a single information and IT architecture***" [20].

In this definition, an Enterprise System could include enterprise resource planning (ERP), customer relationship management (CRM), supply chain management (SCM), and e-procurement systems [21]. The terms Enterprise system and ERP system are interchangeably in the context of this study.

An Enterprise System could include enterprise resource planning (ERP), customer relationship management (CRM), supply chain management (SCM), and e-procurement systems.

2.1.1 Development

The development of enterprise systems was an inside-out process of evolution starting from standard inventory control (IC) packages, to material requirements planning (MRP), manufacturing resource planning (MRP II), which expanded to include other enterprise processes such as sales and order management, marketing, purchasing, warehouse management, financial and managerial accounting (finance), and human resource management. The evolution to extended-ERP systems continued, with the including of inter-organizational processes such as supplier and customer relation management [16].

Like many other technological advances, ERP systems were initially implemented mostly at large organizations. Their relative absence from SMEs has probably been the main reason for the research focus on large companies.

More recently, however, vendors began to provide SME-specific systems. ES adoption at SMEs has been catching up with large companies. Enterprise systems (especially ERP systems) have become so extensive that they are considered the price of entry for running a business and being connected to other enterprises in a network economy [16].

The development of enterprise systems consisting of different components working together was named by Kumar & van Hilligersbergen [16] in 2000 as a possible solution to facilitate the further growth in ES use and development. The development of (business) software as apps did take place and has made designing an enterprise system a lot more flexible and easier (i.e. cheaper) to maintain. This flexibility and cost reduction has opened up the market to smaller companies wanting to adopt an ES, making the distribution of the software a next important factor in the evolution of the systems.

In this software distribution model customers are able to *hire* software applications and use them on demand in a ‘pay-as-you-go’ fashion or through predetermined time subscriptions.

The next big step in this evolution of the distribution of software started with the introduction of the concept of Software-As-A-Service (SAAS). The SAAS model evolved from the application service provider (ASP) model, which emerged in the late 1990s, but did not take off as predicted by analysts. A key issue surrounding ASP adoption became the degree of customization desired by the client and the resulting efficiency loss by the vendor. The doomed ASP model was soon reinvented into the SAAS model, which relied on a different architecture. SAAS is a software deployment model where software or applications are hosted by a vendor or provider to a customer over a network (‘the cloud’). In this software

distribution model customers are able to *hire* software applications and use them on demand in a 'pay-as-you-go' fashion or through predetermined time subscriptions.

This architecture has three important implications: First, it constrains clients' options for customization of the main functionality and data structures of the software. Second, the SAAS model gives more control over future development to the vendor as clients have no choice but to adopt future upgrades of software if they continue using the service. Third, the architecture of SAAS allows for the separation of maintenance responsibilities between the SAAS vendor and the client. [22]

2.1.2 Impact of the SME context

Small and Medium Enterprises (SMEs) have a significantly different context compared to the larger companies. Parts of this context are the characteristics of these SME companies. Zach [23] did research on these specific SME characteristics and developed a framework for them. In this framework the SME characteristics are grouped according to the three contextual dimensions of the TOE framework developed by Tornatzky and Fleischer [24]: organizational characteristics, environmental characteristics, and IS characteristics.

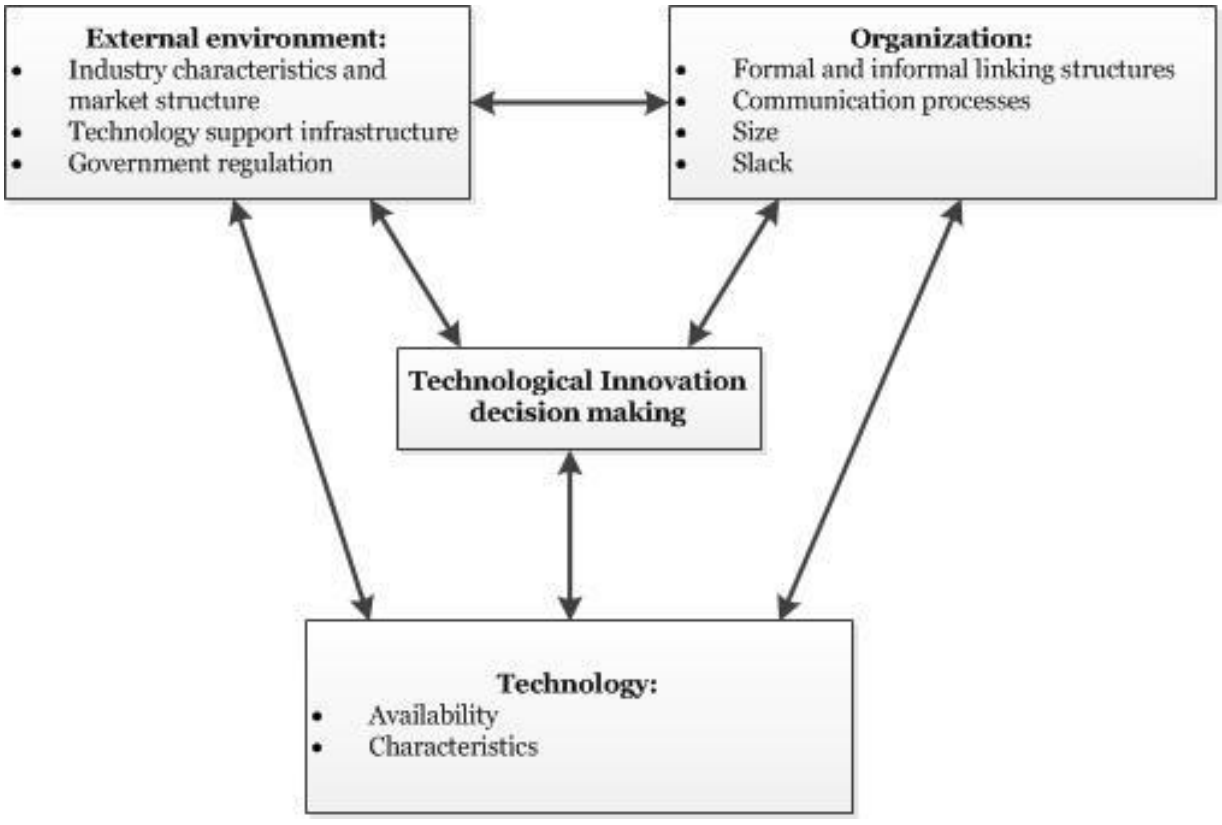


Figure 3: The TOE framework by Tornatzky and Fleischer

Several of these characteristics are common to all small businesses. A small business usually has a specialized product in a small portion of the market as opposed to a wide portfolio of products in a diverse market. The driving imperative of a small business is to deliver the product or service to the marketplace as rapidly as possible, usually at the expense of standardized business processes, which leads to a range of unique business practices. The small business may be extremely susceptible to external market forces like changes in the competitive environment, macroeconomic situation, mergers and acquisitions, and regulatory environment. Revisions to even a few customer orders may significantly impact sales, leading to a volatile cash flow [25]. These factors lead to a demand for subscription based, highly customizable software.

SMEs have been reported having limited IS knowledge, as there is usually insufficient managerial expertise available to plan, organize, and direct the use of information resources. Traditionally, most CEOs in SMEs focus on management issues and pay less attention to technology. The lack of IS knowledge may lead to insufficient attention by management to IS and in turn to a lack of strategic planning of IS implementation and use. In a similar vein, a recent study assessing ERP adoption in SMEs concluded that lack of IS knowledge may inhibit SMEs from adopting ERP systems [26]. The findings showed that the more IS knowledge CEOs have, the more they are inclined to adopt Enterprise systems. Also the results by Chang et al [27] indicate a positive influence of the CEO's IS knowledge as well as employees' IS knowledge on ERP system adoption.

In summary, the resources such as time, finance, and expertise, that are necessary for planning, represent the most critical difficulties in small businesses. Inadequate resources spent on the implementation increase the risk of the failure of the ES implementation.

Table 1: Impact of the SME context on ES implementation capability

Impact of the SME context on ES implementation capability (factors)			
Low IS knowledge	Limited management attention to IS	Lack of strategic planning for IS	Modest managerial expertise
Limited IT/IS in-house technical expertise	Emphasis on packaged applications	Greater reliance on third party	IS function in earlier stages
Subordinated in the accounting function	Limited resources for training	Limited financial resources	Limited human capital

2.1.3 Enterprise Software as a Service

Recently, service orientation has emerged as an important change driver in private and public sector companies. Service orientation offers means to radically improve customer service, business processes and sourcing of information systems. Suppliers have begun to offer information systems according to service based business models, such as Software as a Service and cloud computing. These service based business models challenge the conventional payment models like the on premise installations and design of standard application packages, and have implications for both users and suppliers of these systems.

Through service based business models, suppliers can expand their potential customer base and offer more choices that enable customers to focus on core competencies and reduce initial investments in standard applications. This is a much better fit for the context that SMEs operate in than the conventional business models, making the adoption of an enterprise system less complex and easier to manage for the implementing company.

2.1.3.1 The SAAS model

With the Software as a Service (SAAS) model a customer contracts the use of an enterprise system, such as ERP or CRM, hosted by a third party, rather than buying a software license and installing the application on its own machines [28]. SAAS has many advantages for the client as well as for the vendor. The most recalled advantages of SAAS for clients are the lack of initial investment, faster implementation, no installation, no hardware issues for the client, and no maintenance processes [29].

With Software as a Service (SAAS), a customer contracts the use of an enterprise system, such as ERP or CRM, hosted by a third party, rather than buying a software license and installing the application on its own machines [28].

2.1.3.2 Benefits and risks

For the vendors the advantages that are mostly named are the more predictable and stable income per month [30]. Other advantages are the economies of scale related to the maintenance of the system, investment in the hardware and infrastructure and the costs of development [29]. In a case study where a company was migrating from an on-site enterprise system to the cloud (IAAS or Infrastructure-as-a-Service) done by Khajeh-Hosseini [31], the advantages and disadvantages of the change to cloud based software have also been described. Examples are improved status for using the cloud solution, opportunity to manage income & outgoings, opportunity to offer new products / services, and opportunity to develop new skills.

Gonçalves and Ballon [32] mentioned additional benefits and risks for clients and vendors. For clients the additional benefits are lower total cost of ownership (TCO), eliminate software management and costs become direct costs, focus on core business, always up to date, backup and recovery is handled by vendor and predictability of IT costs.

The important risks mentioned for clients are exposure of business critical data, vendor lock in, loss of data in case of bankruptcy of vendor and less tailoring software [33]. For vendors the risks are high infrastructure investment and development of new skills, the need for contractual expertise, information sharing between multiple applications, and indirect influence upon system and security.

Table 2: Benefits of SAAS

Benefits of SAAS for the customer	**Benefits of SAAS for the vendor**
Opportunity to manage income & outgoings	More predictable and stable income
Opportunity to offer new products / services	Economies of scale related to the maintenance of the system
Improved status	Economies of scale related to the hardware
Removal of tedious work	Economies of scale related to the infrastructure
Improve satisfaction of work	Lower cost of development
Opportunity to develop new skills	
Opportunity for organizational growth	

ERP-as-a-service is ERP delivered through a SAAS model [27]. The main characteristics that distinguish ERP-as-a-service from other types of SAAS models are related to the content of the service. In ERP-as-a service, the service includes (elementary) offerings for enterprise-wide, integrated and standardized business functions and support for business processes which are characteristic to ERP or other Enterprise systems. Essentially it is an *ERP application delivered as a service*, which is accessed through a web browser. In addition to business functionality, the technical infrastructure, the right to use the service, hosting, maintenance and support services are bundled into a single service. The ownership of the software is separated from its use and stays with the vendor. The implementation of such a system is discussed in the following chapter, starting out with the classical on-site implementation process.

2.2 IMPLEMENTING A WEB-BASED ENTERPRISE SYSTEM

The perceptions on the term 'implementation' vary in literature. From a technological diffusion perspective, generic IT implementation can be defined as: "an organizational effort directed toward diffusing appropriate information technology within a user community" [34]. ES implementation studies have employed various stage models, representing the life-cycle, in order to investigate the Enterprise system implementation. The term implementation has been used both to denote the complete process, and a limited part of these ES life-cycle models (e.g. phase three in the framework by Esteves and Pastor [35], or phase two in the model by Markus and Tanis [19]).

With the term 'implementation' in this study, *the phase concerning the implementation and embedding of the system into the organization is meant*, until the system goes live. When the entire process of choosing, implementing, stabilizing, and improving is meant, this will be stated specifically.

Ko et al [36] state that: "As contrasted with more traditional information systems, the [complex information…] systems require understanding and learning by clients that is sufficient for application rather than the lower expectation of system use that has been prevalent in traditional models such as the technology acceptance model by Davis et al [37]." ES implementation requires knowledge of activities associated with configuring and testing ES modules, and training employees in preparation for ongoing operation, maintenance, and support of a vendor-supplied system that is somewhat customized [36].

> ES implementation requires knowledge of activities associated with configuring and testing ERP modules, and training employees in preparation for ongoing operation, maintenance, and support. [36]

In an (Enterprise) system implementation are traditionally three major parties involved: the organization implementing the system (the implementer), the organization that developed the system (vendor), and an organization aiding the implementation (the consultant). Each of these three parties contributes in different ways to the project.

2.2.1 Stakeholders during an implementation

The **implementer** (the organization and its employees) has the detailed knowledge of its own particular business processes, organizational context, and competitive situation, which is essential for successful implementation. The **vendors** provide the implementer with software and offer training programs in connection with their products.

The **consultants** are brought into ES implementation projects to provide additional skills, knowledge, or simply manpower that is not available at the implementer or the vendor, or is too expensive if procured from the vendor. This type of knowledge is typically detailed knowledge of the hardware, software, and implementation process.

It includes knowledge of how to configure the software to meet business requirement needs, as well as organizational change expertise when business processes will need to be changed [38]. The focus of this study is to find ways to bring the expertise and knowledge brought in by the consultants and vendors to the implementing company, making it possible for them to achieve a successful implementation on their own.

The research by Ko [36], Xu & Ma [39], Haines [38], and many others, suggests that in order to implement and operate Enterprise systems effectively, a training model including an extensive training period, and often utilizing external consultants, is a necessary condition.

However, Koh et al [40] found that implementing an Enterprise system does not always require this extensive training. During an in-depth case study conducted at a UK-based SME-specific ERP systems vendor, they found that the SME-specific ERP systems can be implemented and operated effectively by the implementing company with only five days of formal training and no additional consultancy. They state that "It is possible to implement ERP without the use of external consultancy, but that to achieve this; these clients must have exceptional strengths in other areas [40]".

These strengths are to be made possible via the use of a tool, which supports the activities during the implementation stage of an enterprise system (configuring and testing ERP modules, installing software, and training employees in preparation for ongoing operation, maintenance, and support).

2.2.2 Implementation types

Literature has described several types of implementations (or 'transition techniques'), each with different advantages and disadvantages. The types will be shortly described below. They will not play a big part in this research, since the focus of the research is on one particular type, but the characteristics of this type will.

The table below is adapted from Malhotra [9]. In general, there are four basic transition techniques: big bang, phased, parallel, and process line [41]. This research focuses on the Big Bang type of implementation, since this is the type of implementation that has the simplest process: when the new system goes live, all corresponding legacy systems go offline at the same time.

Table 3: Types of implementation techniques

Type	Advantages	Disadvantages
Big Bang *All the functional modules of the new system go live at the same time the legacy system is taken offline*	Costs are reduced, since no interface programs are required. Decision making is simplified; this technique creates a strong focus among the project team members	The go live event requires extensive support; high failure rates are common
Phased *One functional module at a time is transitioned in sequential order*	Companies feel comfortable implementing one module at a time; the resources needed at any given time are low	Additional technical resources are required to develop interface programs to keep both of the ERP systems functional. The transition takes a long time
Parallel *Both the legacy and the new ERP systems operate in parallel for a certain length of time, even up to the go live date*	Good recovery options are available if anything goes wrong with the new system	Considerable more resources are consumed as two systems must be maintained in parallel
Process line *The new ERP system is implemented in big bang fashion, but only one process line at a time*	The experience gained from doing one process line at a time benefits the next implementation	Maintaining communication on both process lines, the legacy and the new system respectively adds complexity

2.2.3 Stages and activities during an implementation

There are multiple theories concerning the different stages during an implementation of an ES. These theories typically consist of a sequence of stages or phases, which depict the stages during an implementation. There are theories that target the general implementation of an information system, like Cooper & Zmud [34], while other theories focus specifically on ERP implementation. These theories will be discussed briefly in this study, since the focus is not on the theories themselves, but to provide the argumentation behind the theory that has been adopted in this study. The various models themselves are included in the appendices at the end of this thesis; the adopted model is also included here.

Parr and Shanks [42] used a model consisting of three phases: Planning, Project & Enhancement. Chang et al [43] used five phases to describe the life cycle of an ERP system. Ross and Vitale [44] had developed a similar model in 2000, indicating the progress after each phase on a timeline. A model using six phases was introduced by Esteves and Pastor [35]. This 6-stage ERP life-cycle framework contains the following phases: Adoption Decision, Acquisition, Implementation, Use and Maintenance, Evolution and Retirement. Each phase is mapped with different activities and issues that are typical for the corresponding phase. The model's focus is on the activities after the implementation, making it less suitable for this study.

The model used in this study is the model by Markus and Tanis [19] [45]. The model consists of four phases, characterized by key players, typical activities, characteristic problems, appropriate performance metrics, and range of possible outcomes. The phases are: project chartering, the project, shakedown, and the onward and upward phase.

- Project chartering includes the activities before the official start of the project. These include the organizational decision about an investment in a new IS solution, mapping of existing business processes, analysis of potential benefits and limitations, specification of functionality needed, and system selection.
- The project phase encompasses all activities between the system selection and "going- live". It comprises activities such as project team building, business process modeling and reengineering, system customization and configuration, end user training, data conversion, testing and debugging, and rollout.
- Shakedown is defined as the period between "going-live" and the time when operations get into routine use. During this phase the system performance is tuned, bugs are fixed, and additional training is conducted if needed. The end users are getting familiar with the system and operations are becoming "normal".
- The onward and upward phase is defined as the period since "normal" operations to when the system is replaced by an upgraded version or a different system. Typical activities involved are additional user skill building, continuous business improvement, and benefits and success assessment.

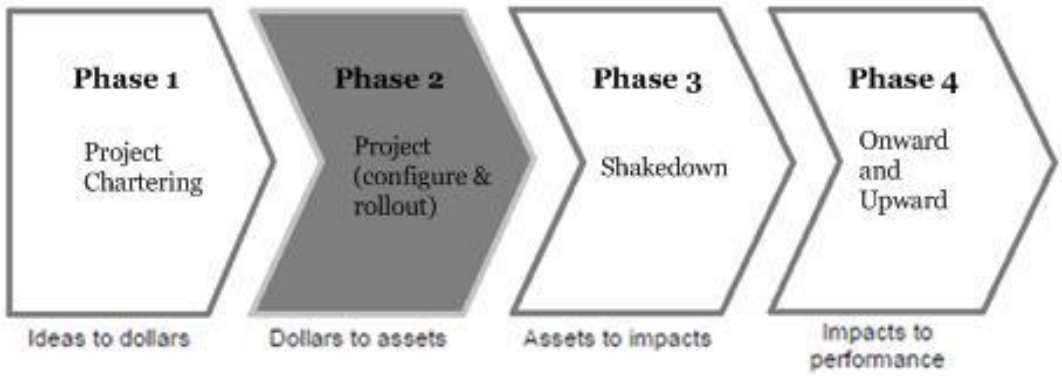


Figure 4: Model adopted from Markus and Tanis [15], to show the focus of the research

2.2.3.1 Activities during the 'Project' stage

The stage of the Markus and Tanis model that will be used in this thesis is the Project phase. During this project phase the application is configured, integrated and at the end ready for use by the end users. The activities during this stage are the key activities of this study. When the implementing company is able to successfully complete these tasks themselves, the implementation can be done by the company itself.

The research of Markus and Tanis considering the project phase is summed up in the table below. This table was adopted from their study [19]. Important parts are the activities and the errors and problems found. These errors and problems have to be minimized in order for the implementing company to be able to do the necessary activities without external guidance. Other information that will have to be found during this research is the priority and length of the activities, as these are needed in the selection of activities that are risks and need additional support (from the prototype) during the implementation.

These activities and risks are based on an *on-site* implementation; the server with the application is installed at the implementing company and is hard to reach for the external consultants and vendor.

The context of this study is an environment that has adopted the principle of software as a service: the software is delivered through the internet, making on-site installation obsolete and giving the vendor (and consultants, if needed) better access to the application. This difference in context should have an impact on the activities, errors and problems. The literature found on implementation of an ES in a SAAS environment is described in the next chapter of this research.

Table 4: The Project phase of an implementation

Project – Configuration, integration, and rollout				
Description			**Key actors**	
Activities designed to get the system up and running in one or more organizational units			Project manager, project team members, and a variety of external technical and general management consulting resources, executives (in steering committee capacity), other organizational members (in consultative roles)	
Typical activities				
Development of a detailed project plan	Training of project team members and acquisition of supportive skills	Current and/or future business process modeling and reengineering, if any	Integration of software bolt-ons and/or legacy systems, if any	Selection and assignment of project team members
Ongoing project management	Software configuration	Software customization, if any	Data cleanup and conversion	Executive and end-user training
Testing, bug fixing, and rework	Execution of change management plan, if any	System integration	Documentation	Rollout and startup
Common errors and problems				
Staffing project sub teams without appropriate cross-functional representation	Difficulty acquiring adequate knowledge and skill in software configuration, integration of bolt-ons or legacy systems	Poor-quality software, documentation, training materials	Inadequate knowledge on part of consultants and vendor personnel	Configuring software for multiple units on the basis of analyzing only one unit
Assuming that end-user training should be funded from operations budgets	Configuration errors that require rework if caught	Customizations that do not work	Failure to manage project scope, schedule, budget	Inadequate attention to data cleanup

2.2.4 Implementation in a SAAS environment

The difference between the implementation of on premise ES, which are regarded as traditional implementations where the system is hosted on-site and the company purchases and "owns" software and software licenses, and SAAS on the other hand, is that it is deployed in a model that allows for the provision and use of an Enterprise system by a vendor or provider via the Internet. In this case, software applications are hosted and provided by a vendor on a subscription or lease model over a network, which typically is the Internet.

> [SAAS] applications are hosted and provided by a vendor on a subscription or lease model over a network, which typically is the Internet

Implementation of ERP in a SAAS model is an important factor amongst customers, 10 experts had expressions about complexity regarding implementation, such as: "the ERP implementation effort has lost its complexity". This reduces the allocation of capital and allows for faster startup of ERP [20]. Another added that the "proof of concepts, testing as well short development cycles are greatly enhanced" which leads to low implementation costs and the rental cost model lead to a reduced vendor lock-in. All participants stated as main reasons behind adopting SAAS is that "the risk of a possible bad implementation shifts from the customer to the provider" in line with traditional outsourcing of IT [21, 22].

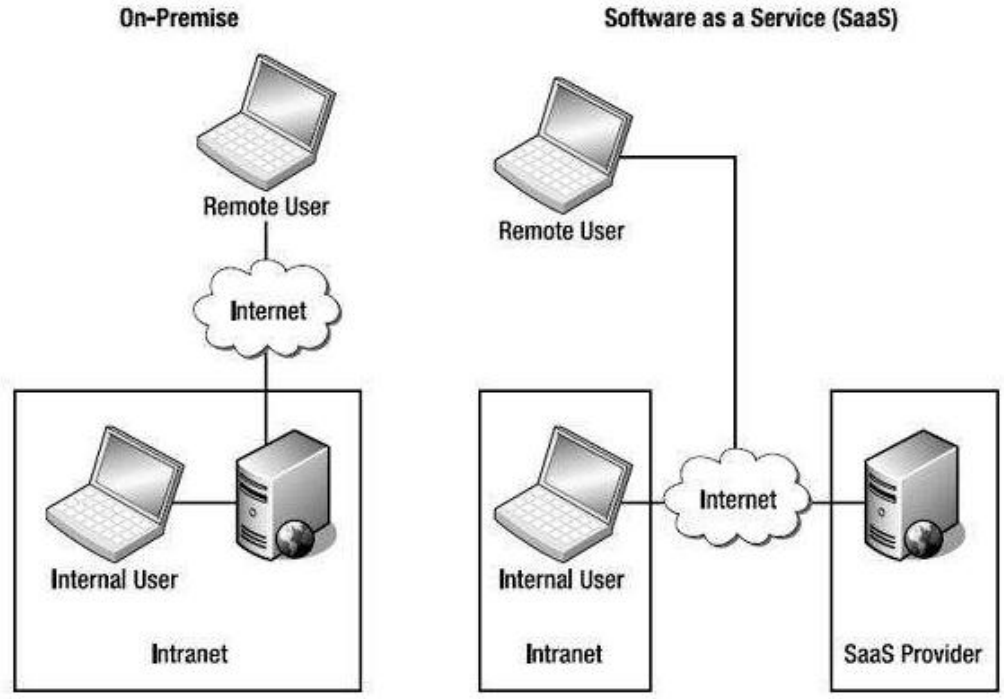


Figure 5: The difference between On-Premise and SAAS

2.2.5 Effort during implementation

Francalanci [46] found in her research on implementation effort during the implementation of SAP/R3 at multiple companies in 2001, that the technical size of software is not sufficient for predicting the implementation effort of an ERP system accurately, while organizational measures of project complexity are also critical drivers of effort. The positive correlation between effort and organizational size and number of users indicates that a package requires increasing resources in order to be implemented in larger companies, which should indicate that smaller companies (SMEs) require less effort.

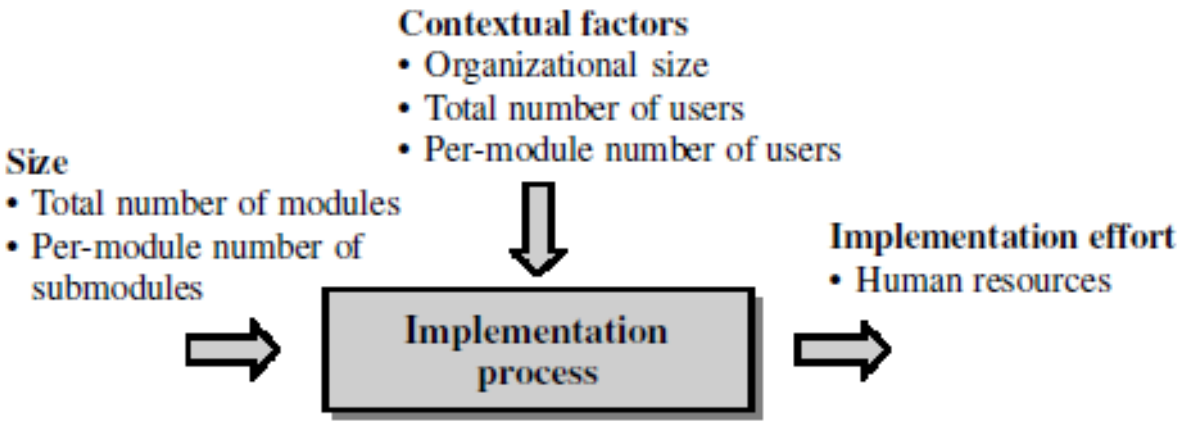


Figure 6: Effort during implementation

The definition of 'implementation effort' for the ES project used in this research was operationalized as the total man hours for the project management, operating teams and functional units. In this paper the implementation effort will be described as the total amount of hours the implementing company needs.

2.2.6 User participation during implementation

There are two main areas of user participation when a company or organization decides to implement an Enterprise system (Esteves et. al. [13]). The first area is when a user participates in the stage of definition of the company's ES needs and the second area is user participates in the implementation of the ES (Zhang et al. [33]).

Understanding the contributions of user participation in ES implementation will lead to successful business implementation.

2.3 PROTOTYPE MODELING

The recommendations that follow from the literature, complemented with the data from the interviews, will be used to design a prototype that will be tested in an experiment. In this experiment the practical implications of the research are tested (the answer provided to sub question 4) by selecting SME business owners and asking them to design and implement their own enterprise system based on their organization using a tool developed from the recommendations that follow from the sub questions.

To be able to test the model that will be developed through the literature, literature on building a prototype and guiding the persons taking part in the experiment is needed. This literature will be discussed in the following chapters, starting with the building of the prototype. By using literature on this topic, the validity of the (prototype used in the) experiment will be as high as possible, leading to more valid results.

The main approach to this is the use of the Design Science process iteration model, as stated by Peffers [17].

2.3.1 Building a prototype

The design of the prototype can follow many different paths. The design process of information systems has different theoretical grounds to draw from. More theories on the design of a prototype according to a model can be found in the design science literature. After the theoretical model is made, the research follows shall follow a design science approach to complete and validate the prototype. The methodology behind this approach is taken from Peffers [17] and is discussed at the chapter describing the design of the prototype (chapter 4.1). The experiment will use a ‘path’ that is compiled from the application that the customer has assembled. This path will cover all the remaining implementing activities and will try to support the implementing company in such a way that the implementation of the enterprise system can be considered a success. To be able to guide the implementer the prototype will have to provide steps, which follow the (ideal) implementation process. This process is detailed in chapter 3.

2.3.2 Gamification

An interesting way to keep the employees of the implementing company interested in following the prototype is gamification. Gamification is the use of gaming concepts during a non-gaming process [47], to increase immersion and motivation to complete the process. The gamification concept can be seen frequently nowadays, in many different kinds of context. Hamari et al. [48] found in their review that gamification has a positive effect on motivation and engagement in learning tasks, as well as enjoyment over them.

2.4 CONCLUSION

The literature discussed in the chapters above can be divided into three main factors that influence the ability to implement a system: the implementing organization's context, or business architecture (Zach [23]), the technical architecture (web based or on site), and the system architecture (Koh et al [40]).

1. The context of the implementing organization is related to the amount of resources available, competitive pressures, and the amount of IS knowledge. These directly influence the ability of the company to implement the system themselves.
2. The architecture of the system influences the activities (in complexity and amount). In a SAAS environment, most of the architecture is already in place, and the software is rented from a pre-installed server. This greatly affects the implementing activities, since the activities like testing, bug fixing and software customization are done by the vendor.
3. The design of the system is the last main factor influencing the ability to implement. Ease of use and a generic design of the functional components of the system lower the amount of training required to use the system. The use of effective support tools during and after implementation can further reduce the need of training.
4. The system architecture and the system design influence the implementing activities and the amount of training that is required to fully use the enterprise system.

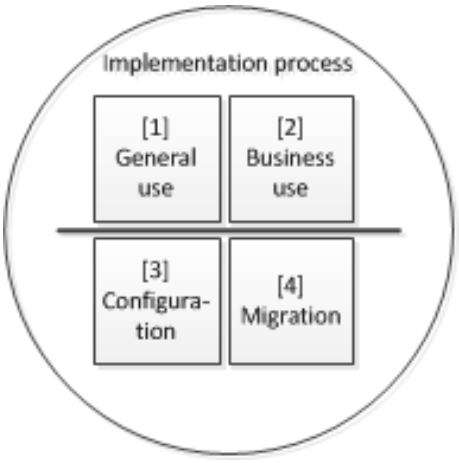


Figure 7: The implementation process and activities

The literature research results can be represented using the model shown above. This model is the embodiment of the implementation process of a SAAS based enterprise system. The implementation process can be seen as the top layer that provides the structure for the various implementing activities. These activities can be divided into activities concerning the general use of the system (1), concerning the training in the business processes included in the system (2), concerning the configuration of the system (3), and the conversion and migration of the existing data (4).

The implementing activities can be divided into two levels: the business level, with the general use and business processes, and the technical level, with the configuration and migration of the system. The four implementing activities will be discussed in the following chapters and are at the core of the tool.

3 THE IMPLEMENTATION PROCESS

In this chapter the literature review from chapter 2 will be assembled into an integrative model. This model will explain the connections between the factors that influence the ability of the implementing company to implement its own system and create a model that combines these factors.

The model of the implementation process by Markus and Tanis will be used as the starting point for the model. The project phase of this model describes the implementing activities that are needed. To find the activities that are still relevant in a web based implementation, interviews with consultants at Novulo have been conducted. These interviews revealed the remaining activities and are discussed in chapter 3.1. The findings have been incorporated into the model by Markus and Tanis to create the web based implementation activities. The activities have been designed into a process and this process will be shown using the Bizagi modeler in chapter 3.2.

After the implementation process has been defined and the activities have been further clarified, the model itself will be discussed. The model is based on the literature found and combines the theories of Markus and Tanis, Koh et al., Zach, and Francalanci into a single model describing the implementation process and the factors that influence the ability of the implementing company to complete the activities during the implementation of a web-based enterprise system.

3.1 DEFINING THE IMPLEMENTATION PROCESS

The implementation process model that is used in this thesis is the IS life cycle model by Markus and Tanis [19]. Their model tries to explain the activities related to the implementing process. The model has been developed with an on-site implementation in mind, so not all activities will be relevant in a web-based implementation. This difference hasn't been researched in the literature yet, making it necessary to use a different way than literature research to find the information needed.

To find this difference, interviews have been held with the consultants at Novulo, asking them what activities were still relevant in a SAAS environment. This chapter will discuss the results of these interviews, starting with the original model and activities by Markus and Tanis. The argumentation behind the selection of the remaining activities is explained by using quotes from the interviews, with the remaining activities being described in the conclusion of the chapter.

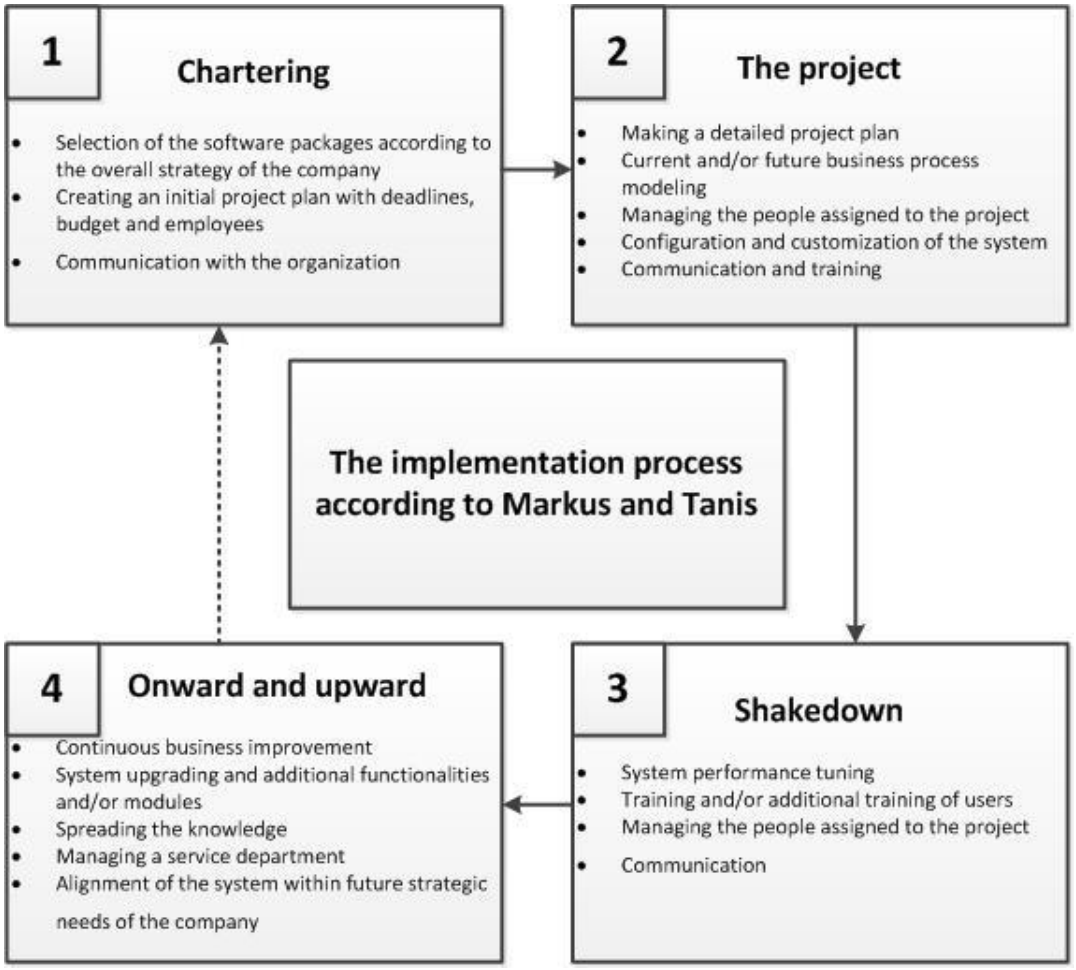


Figure 8: The implementation process mapped out

Each of these implementation steps has their own activities. Not all activities are relevant however for this research. The Chartering phase has already been completed by selecting the SAAS software supplier, and the Shakedown and Onward and upward phases can only be completed after the initial implementation of the system. The focus of the research is on the Project phase, the phase up until the actual use of the system. This phase has certain activities that need to be completed to make the system ready for use.

3.1.1 Implementation activities

The activities that remained according to the interviewed consultants are explained in more detail below. These activities will be the activities that have to be taken care of during the implementation of a web based system and being able to complete them will define the ability of the implementing company to implement its own system. The other activities have no further use, since they are done by the software supplier or have become irrelevant. The remaining activities are listed below. The motivation of the consultants is given in the appendices.

Table 5: Activities during the Project phase

Execution of change management plan, if any	Training of project team members and acquisition of supportive skills	Current and/or future business process modeling and reengineering, if any	Integration of software bolt-ons and/or legacy systems, if any	Selection and assignment of project team members
Ongoing project management	Software configuration	Software customization, if any	Data cleanup and conversion	Executive and end-user training
Testing, bug fixing, and rework	Development of a detailed project plan	System integration	Documentation	Rollout and startup

3.1.2 Remaining activities

The activities that remain after eliminating the activities that have become irrelevant in a web based implementation are listed below. These activities will become the core activities that have to be supported by the model and the practical experiment that follows from it.

A detailed explanation of the activities is required to provide the insight needed to inform the implementing company about the work that is required to complete the implementation process (or Project phase). The argumentation is again provided in the appendices.

Table 6: Remaining activities during a SAAS implementation

Development of a detailed project plan	Current and/or future business process modeling and reengineering, if any	Integration of software bolt-ons and/or legacy systems, if any
Software configuration	Data cleanup and conversion	Executive and end-user training

3.2 THE IMPLEMENTATION PROCESS

In a web based implementation certain activities that are needed have already been pre-installed by the vendor. This makes implementing a web-based system a lot easier and faster than the classical on-site implementation. The entire implementation process is depicted below, with the Project phase containing the activities that remain after the interviews with the consultants.

The modeling of the implementation process has been done with the Bizagi Modeler software. The model is divided into the four phases by Markus and Tanis that have been discussed before. The activities and decisions have been implemented into the process, giving an as complete as possible image of the implementation steps. The activities are divided between the implementing company and the consultant/vendor (software supplier). The division is made according to the stakeholders selected by Haines [38]. The implementation of a web based enterprise system does not require a consultant, so the consultant and vendor have been grouped together.

Each phase also has a table included describing the various inputs, outputs, activities, and techniques that are relevant for that phase.

3.2.1 Phase 1: The chartering phase

The chartering phase is mostly focused on the decision making during the selection of the software and the vendor of the enterprise system. The process starts with a company deciding to adopt a (new) enterprise system. After this decision, the most important characteristics and critical success factors (CSF's) are decided and the search for a software vendor starts. When a vendor is found, the bidding process begins and an agreement is reached (or not reached). Part of this agreement is the initial plan, which is usually done with the help of the vendor. When the initial plan is accepted, the project phase begins.

Table 7: Chartering - Inputs, Outputs, Activities

Type	**Explanation**
Inputs	• Ideas
Outputs	• A project plan for the implementation of the chosen application
Activities	1. Define the CSI's 2. Decide to implement the system 3. Form project plan
Techniques	• Decision making techniques, based on the CSI defined in the process • Plan making techniques

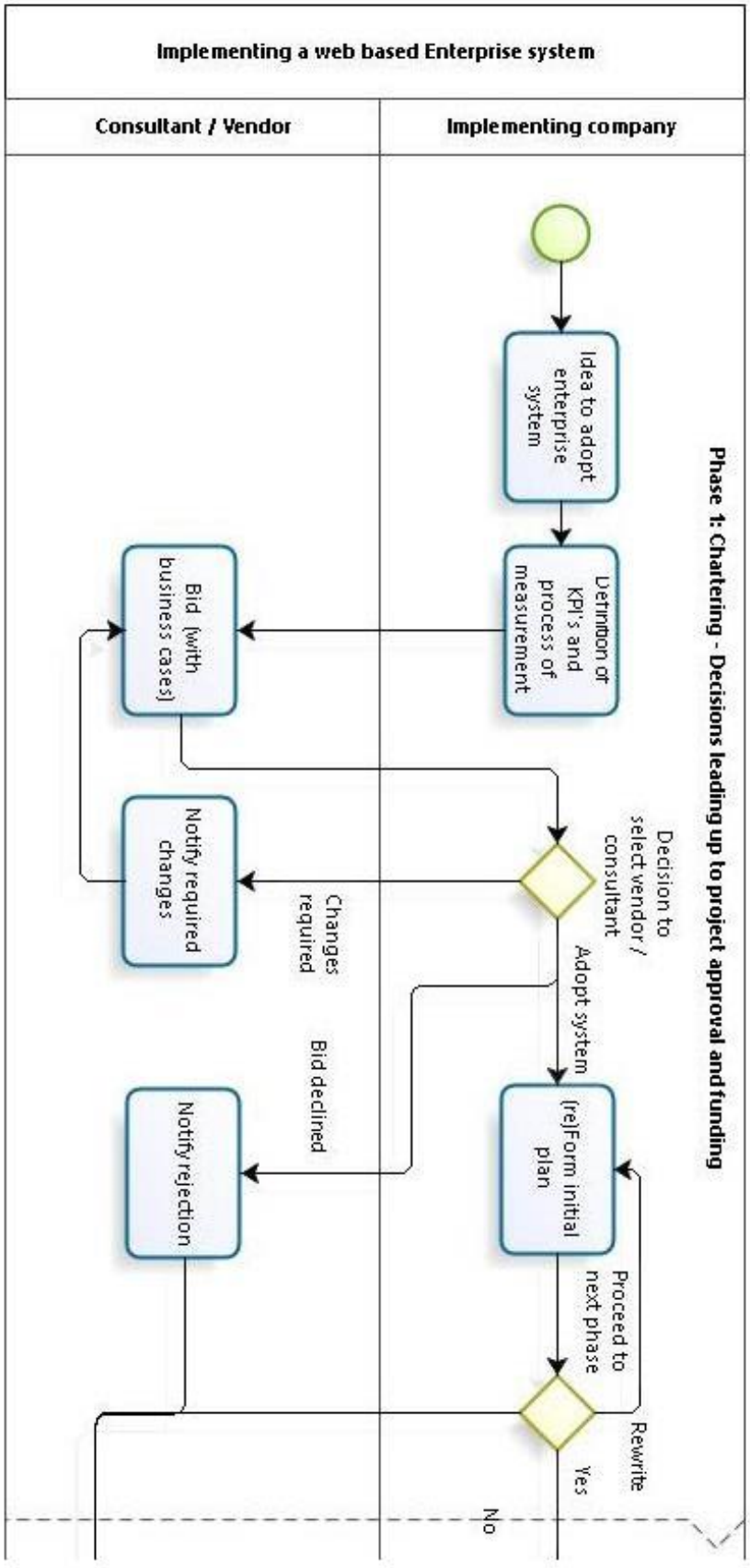


Figure 9: Implementation phases I: The Chartering phase

3.2.2 Phase 2: The project phase

The Project phase is the focus of this research. The phase describes the activities leading up to the actual use of the enterprise system. The activities that remained after the interviews are depicted in the order that they are encountered and start with the selection and assignment of the team members who will implement the system. These members are trained in the use of the application and are led through the business processes that encompass the application via test data. During the training, the software is configured by the implementing team to suit the (specific) needs of the company. A standard configuration is present in the application, providing all the necessary settings needed to have a working application.

When the configuration is in place, the decision on the migration of the legacy data is put into practice. The activities during this stage depend on the type and amount of legacy data, the legacy system itself (brand), and the validity of the legacy data. Most legacy systems have a program that automatically converts the data to the format needed in a Novulo system. Using such a program makes the time needed for the data conversion and migration a lot shorter and decreases the possibility of mistakes. However, developing such programs is time consuming. When there is no conversion program available for the software (or if the legacy data is in Excel), the conversion will have to be done manually, by matching the data between the two systems using Excel sheets (match a row in the database of the legacy system to a row in the new system). This matching requires testing the data, which is done via a test migration.

When the test migration has been completed successfully, and the implementing company has confirmed that all data has been migrated correctly, the final conversion is planned. After this conversion, any alteration to the data in the legacy system will not be migrated towards the new system, so the approval of the functionality of the application has to be given before the final conversion. After the final conversion (usually the same day), the system is ready for use, the normal work processes start in the company, and the next phase begins.

Table 8: Project - Inputs, Outputs, Activities

Type	**Explanation**
Inputs	• Project plan
Outputs	• A fully implemented system, ready for use
Activities	1. Train users 2. Configure application 3. Migrate legacy data 4. Approve implementation
Techniques	• Tools available to support the implementation process • Training instructions • Migration tools / sheets

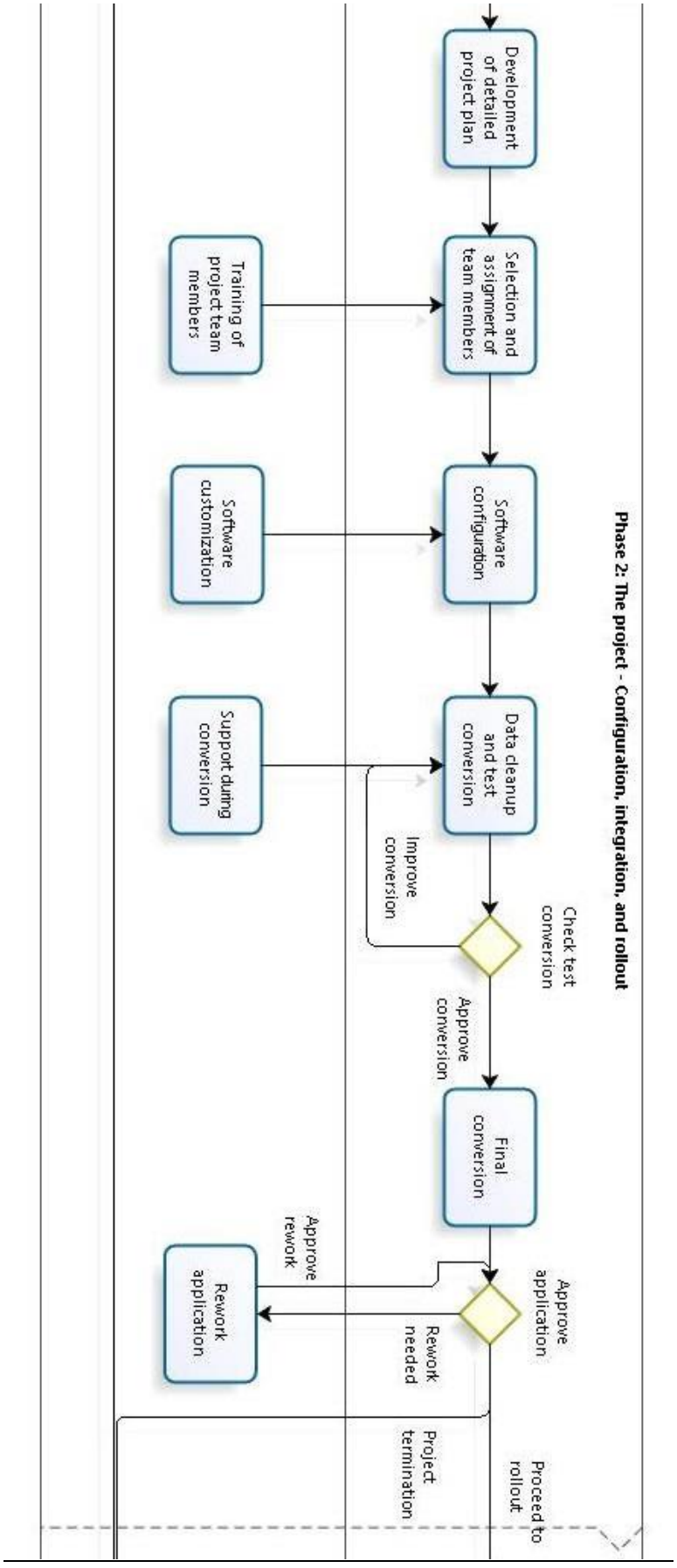


Figure 10: Implementation phase 2: The Project phase

3.2.3 Phase 3 & 4: The Shakedown phase and the Onward and Upward phase

The shakedown phase and the onward and upward phase describe the continuous improvements that are made after the initial rollout. Quality improvement and improvements to the system make up the core of the activities during these phases, with the Shakedown phase ending when normal operation has been achieved.

During the final phase, the system is in use and the company starts to look towards improvements of the system or even adopting a different system. Improvements can be additions to the system, for example by adding functionality, or adapting the system to the needs of the company in certain areas. If the choice is made to adopt a different system, the implementation process starts all over again.

Table 9: Shakedown - Inputs, Outputs, Activities

Type	**Explanation**
Inputs	• A fully implemented system, ready for use
Outputs	• Normal operation
Activities	1. Measure outputs of normal operation 2. Check if output is sufficient
Techniques	• Control/measuring techniques

Table 10: Onward & Upward - Inputs, Outputs, Activities

Type	**Explanation**
Inputs	• Normal operation
Outputs	• Upgrade of system or decision to implement new system
Activities	1. Decision to upgrade or implement new system 2. Continuous improvement
Techniques	• Continuous improvement techniques

Building the process into a tool

This process will have to be supported by the implementation tool that will be designed and used in the experiment. The tool will focus on supporting the implementing company during the project phase. The activities during this phase have been discussed before and the order of these activities has been depicted in the implementation process. The literature on tool-making has been discussed in chapter 2.3 and will be used as guiding principles in the design of the tool.

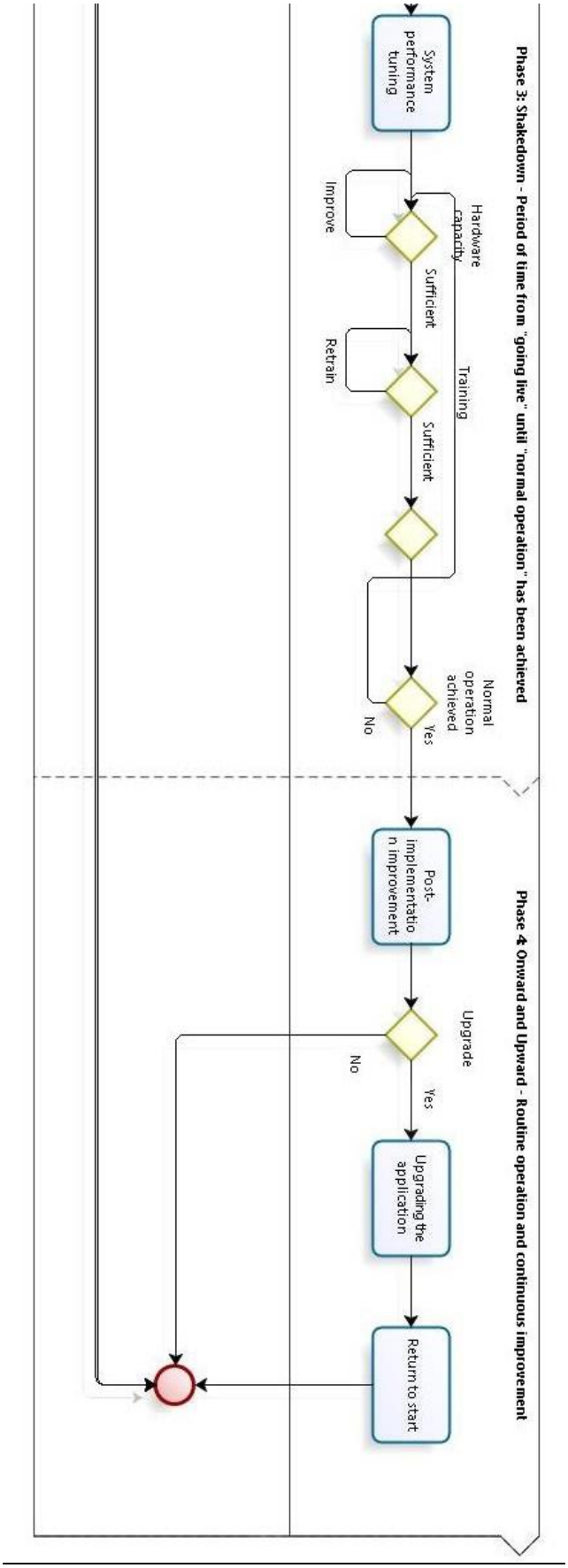


Figure 11: Implementation phases III: The Shakedown & Onward and Upward phases

4 TOOLS ENABLING THE SME IN IMPLEMENTING A WEB-BASED ES

The literature research and the implementation process above lead to the research model below. The model describes the context (the factors influencing the implementation process) and the relations with the implementing activities, training requirement and overall effort needed to successfully complete the implementation of the system. The use of a tool to support the activities and training of the implementing company will have an impact on the ability to implement the system according to the model.

The model consists of three parts, which lead to the ability of the implementing company to implement its own web based enterprise system. The model starts with the general factors that influence the amount of activities and training that are required to implement the system successfully. These factors are adapted from the literature (as seen in the conclusion of the literature research). The organizational context [23], system architecture [22], and the system design [40] were found to be the main factors in this regard.

4.1 THE INTEGRATIVE MODEL EXPLAINED

The **business architecture** (or organizational context) includes the companies' technical, organizational, and environmental characteristics. These characteristics explain the knowledge, resources and restraints within the implementing company. The characteristics can't be altered by the tool, but most definitely have an impact on the ability to implement an enterprise system. In this research, the SME context is used. By specifying the context, the general characteristics of this context can be used, minimizing the effect on the validity of the research results. Further research should be done to find the influence of these characteristics on the implementing ability of the company however.

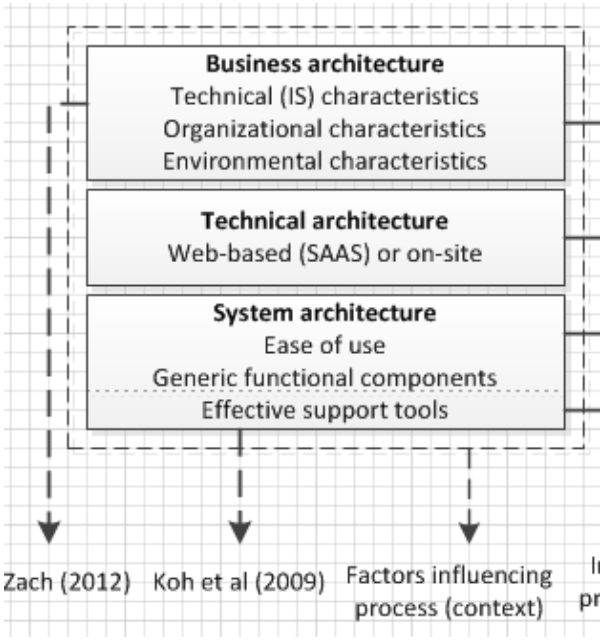


Figure 12: Model - Factors influencing the process

As stated in chapter 3.1, the **technical architecture** of the implemented system has a large impact on the amount of activities that are needed to implement the system. An on-site implementation of an enterprise system has too many activities, some that can only be done by the vendor, making implementation by the implementing company impossible. Implementing a web based (SAAS) enterprise system eliminates many of the activities, creating the possibility to implement your own system.

The **system architecture** has an impact on the amount of training that is required to work with the enterprise system. When the system is easy to use and the processes fit with the current business processes, the amount of time to transfer to the new system is decreased. The use of effective support tools during and after implementation can further reduce the need of training.

The factors of the technical and the system architecture influence the secondary factors: the number of implementing activities and the amount of training required. These secondary factors can be influenced by the tool and can be measured in the experiment.

The **implementing activities** remaining in a SAAS context have been discussed before. Each of these activities will have to be addressed by the tool in the correct order. The order of activities is determined by the reference implementation process. Completing an activity should provide a reward, such as a progress bar to increase motivation (gamification).

The **training** requirement is determined by the ease of use of the software, the amount of BPR required and knowledge within the company.

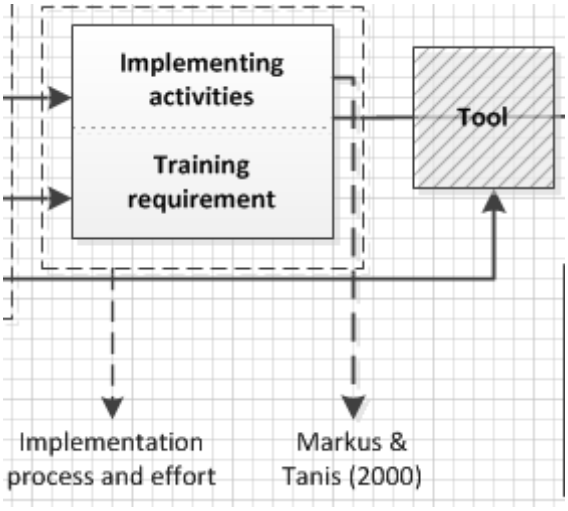


Figure 13: Model - Implementation process

Since the starting knowledge within the company cannot be influenced by the vendor, the other two will remain and have been included in the model. The instructions used in the training is to be included in the tool and should provide the support needed to complete the implementing activities and (after the implementation has been completed) start the daily work processes.

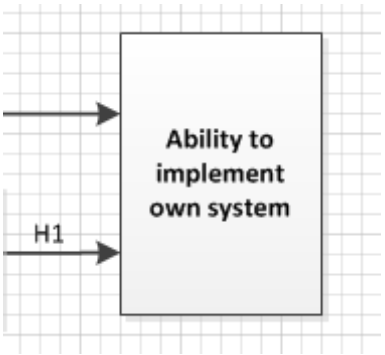


Figure 14: Model - Ability to implement own system

With the ease of use of the system and the fewer remaining activities, it could be possible to implement a system without the use of consultants. The **tool** must ensure that the abilities of the implementing company are sufficient to successfully implement their own system.

This ability can be seen as the amount of activities that the implementing company can **successfully complete** on its own.

The integrated model is depicted below, which shows the all the parts combined. This model is the first result of this study.

4.2 COMBINED RESEARCH MODEL

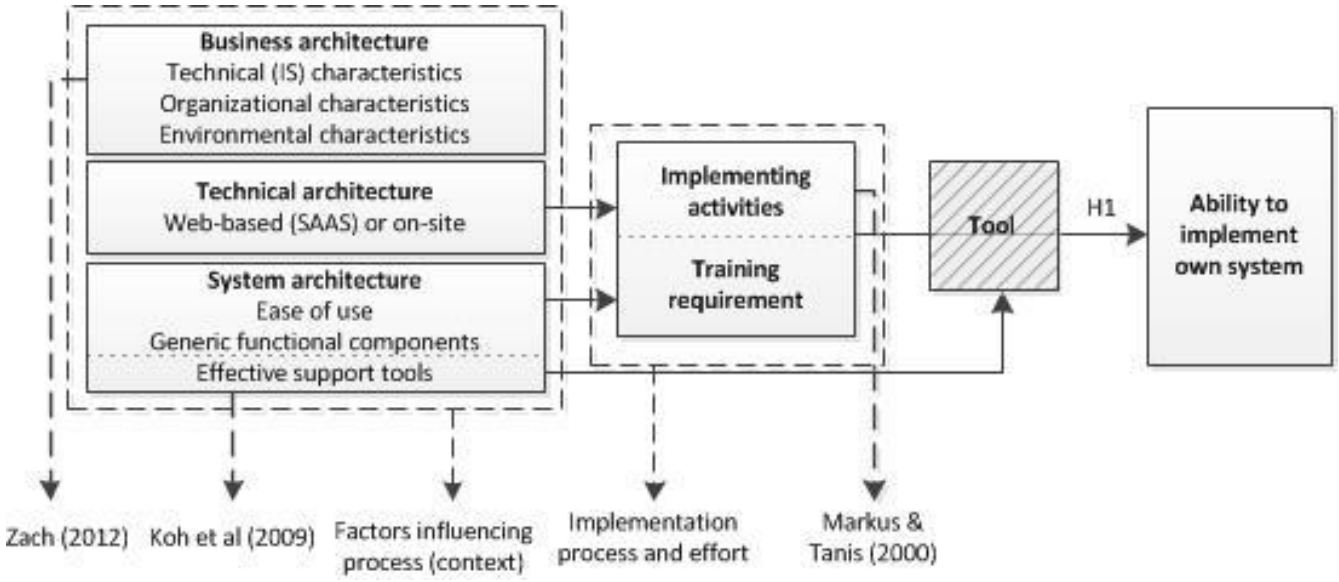


Figure 15: The research model: Tools enabling an SME in implementing a web-based ES

The above research model can be simplified, to ensure easy use in the further research needed in this thesis. This simplified model describes the research model in independent and dependent variables, and the outcome. This model will be used (where needed) in the other two parts of this thesis: the model of a web based implementation process in an SME and the experiment conducted to validate the model.

The independent variables in the model are the combined factors influencing the implementation process. These are the business architecture, technical architecture, and system architecture. The dependent variables describe the implementation process and are the implementing activities, training requirement and the tool. The implementation process eventually leads to an outcome, measured in the amount of activities successfully completed by the implementing company.

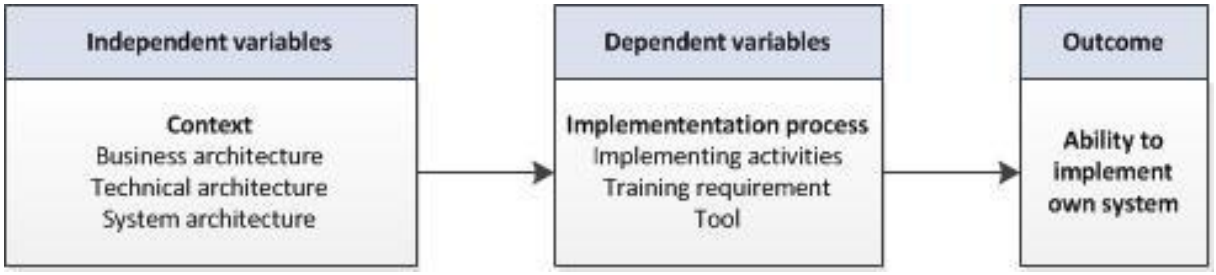


Figure 16: Model - Research model

5 PROTOTYPE

In this chapter the design and development of the prototype used in the experiment is discussed. The prototype is based on the model described in the literature review and will test the possibility of implementation done entirely by the implementing organization. This means that all implementing activities named by Markus and Tanis [19], if relevant for the implementation, will have to be supported by the prototype tool. The activities have been discussed in the literature review and adjusted for the implementation in a SAAS environment. The prototype will be a part of the enterprise system and will be available to the (lead) user at startup.

5.1 GOALS, CONCEPTS AND OPERATIONALIZATION

The goals and the concepts and their operationalization are discussed in this chapter. Their function is to provide insight into the functionality the prototype has to offer. This is crucial during the design, making sure that the conversion from theory to practice is done correctly.

5.1.1 Goals

The main goal of the prototype is to provide all the support a lead user needs to implement a web based enterprise system in their own company. To achieve this, the hypotheses stated in the model will have to be proven as valid. Each of the hypotheses describes an activity of the remaining activities from the model by Markus and Tanis [19].

A specific part of the prototype is aimed at supporting the implementing company through the entire process of the implementation (as shown in chapter 3.2). The knowledge of an implementation of an enterprise system and the activities, risks, and opportunities that arise during such an implementation can have a large impact on the success of the implementation. This means that the prototype should not only support the specific activities themselves, but should also provide an overview of the entire project.

5.1.2 Concepts and operationalization

The model framework provides several factors that influence the ability of the implementing company to implement their own system. They will have to be listed as concepts with definitions and operationalized to be able to measure them. The concepts that will be tested during the experiment and their operationalization are described below.

Table 11: Concepts and operationalization

Concept	Operationalization
Amount of support required	Number of questions asked
Amount of time needed to complete activities	Number of minutes required
Amount of activities completed	Completion of activity (in %)
Satisfaction	Survey rating
Usability	Survey rating

The first three concepts can be directly measured during the experiment. The number of questions asked, minutes required and completion (in %) will all be registered. These numbers represent the effectivity of the prototype to support the business user during the implementation.

The satisfaction and usability surveys address the usefulness of the tool. These ratings provide the validity to the tool, by checking if the user is able to work with and satisfied about the tool.

5.2 DESIGN

The design of the prototype should follow the same rules as the model dictates for the enterprise system itself; it should be easy to use, require no business process reengineering, and have sufficient support within itself. These are the main reasons the prototype has been kept simple in its appearance, there are few options available at first, making it less confusing when the implementing person logs into the application for the first time.

5.2.1 Functional design

The functionality needed to provide the support the implementing company needs according to the model, is the core of the prototype. If the prototype doesn't support all factors, it is likely the implementation would fail. The hypotheses each cover a different implementing activity, making identifying the factor(s) causing the failure easier.

The implementation process can be divided into several functional parts. These parts are the various implementation activities grouped into four subjects that are held together by the implementation process itself. These subjects are described in detail below:

- **Implementation process:** The process that encompasses the entire implementation, from start to finish. Since knowledge about this process is lacking in an SME, the tool should guide the implementing company through this process, with clear directions and milestones. The implementation process will be one of the first overviews to be displayed, showing the implementer the steps to be taken and the time that is required for them in a linear fashion.

- **General:** Instructions in the general use of the application. These are application specific, since every software system has a different User Interface and navigation. Subjects of the integrative model included in this segment are the general system design and use of the application, providing the basic knowledge that is essential for the use of the system.

- **Training:** This subject is based on the training activities that are required during an implementation project. It discusses the various processes a user has to do in their daily routines, like customer, sales, or purchase registration. This part of the implementation tool can also be used for the regular training of the users, adding further value to the tool.

- **Configuration:** Configuration is everything that is related to the set-up of the system. Every that can be configured has to be set up in such a way that the implementing company can work optimally with it. Configuring these settings and checking them usually requires knowledge about the enterprise system.

- **Data:** The data subject handles the conversion and migration of the legacy data, if any. The conversion and migration of the data can be done in several ways, depending on the type and brand of legacy system. Options available are the automatic conversion using a toolset (if available), a manual conversion, or no conversion. This choice can be made on several levels, depending on the desired amount of data that needs to be converted.

5.2.2 Graphical design

The graphical design should provide the implementing company an easy to use, straightforward interface. The design of the User Interface (UI) of the Novulo software is already heavily focused on ease of use, with the Metro functionality. With the Metro functionality, anything can be 'pinned' to the Metro screen (not unlike Windows 8).

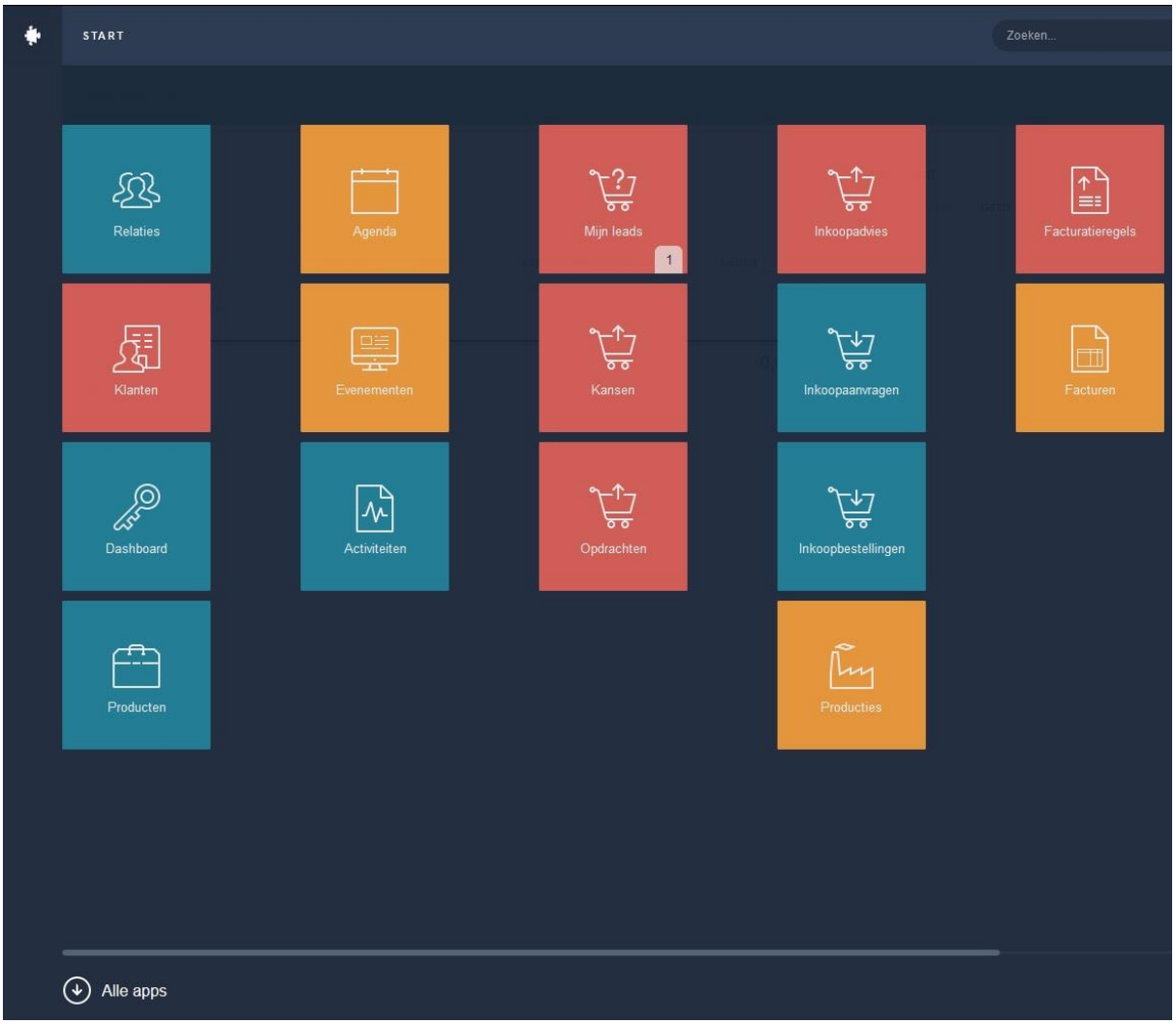


Figure 17: The Metro dashboard

5.3 THE PROTOTYPE

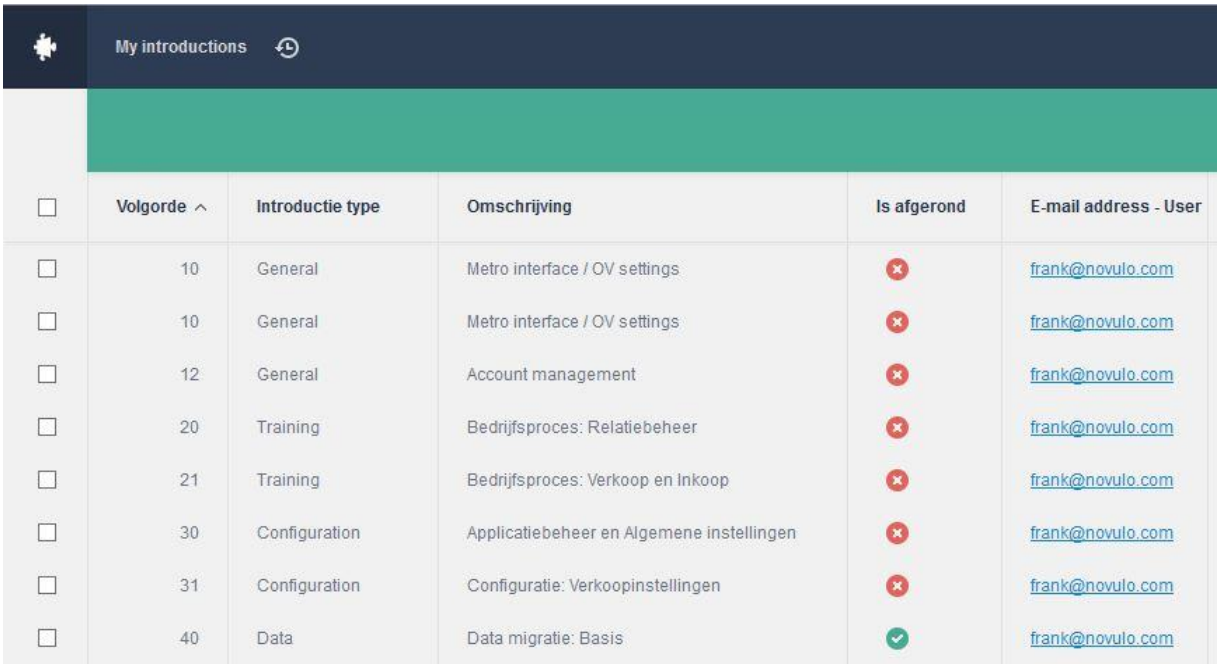
With the goals, concepts and design specified in the chapters above, the design of the prototype is set. Developing the component started with defining the content that should be shown and how it could be shown in an understandable and usable manner. The description of the content can be found in chapter 5.4.1. This chapter is based on the functional design discussed in chapter 5.3.1. After the definition of the content, the interface of the tool will be discussed and the implementation process defined and represented in the tool is explained. This process will be illustrated by describing a case that is representative for the prototype implementation process.

5.3.1 Content

The content of the prototype has to guide the user through the steps of an implementation. This guidance is based on the functional design described in chapter 4.3.1 and uses the subjects to define the content of the tool. The content forms the core of the tool. The content is to be delivered in the form of 'introductions'. Each introduction handles one topic of a certain subject and can be set to a certain user (employee). Some topics are more specific than others, like the overview settings (OV settings).

Table 12: Possible introductions

1. General a. Metro interface / OV settings b. Navigation c. Account management d. Reporting e. Package specific	**2. Training** a. Business process: CRM b. Business process: Sales and purchasing c. Business process: Invoicing d. Business process: Finance e. Business process: Logistics
3. Configuration a. Application management and General settings b. Sales settings c. Purchase settings d. Financial settings	**4. Data** a. Data migration: Basics b. Data migration: Cleanup of legacy data c. Data migration: Conversion



Volgorde	Introductie type	Omschrijving	Is afgerond	E-mail address - User
10	General	Metro interface / OV settings	✗	frank@novulo.com
10	General	Metro interface / OV settings	✗	frank@novulo.com
12	General	Account management	✗	frank@novulo.com
20	Training	Bedrijfsproces: Relatiebeheer	✗	frank@novulo.com
21	Training	Bedrijfsproces: Verkoop en Inkoop	✗	frank@novulo.com
30	Configuration	Applicatiebeheer en Algemene instellingen	✗	frank@novulo.com
31	Configuration	Configuratie: Verkoopinstellingen	✗	frank@novulo.com
40	Data	Data migratie: Basis	✓	frank@novulo.com

Figure 18: Introductions included in the tool (Dutch)

5.3.2 Interface

The prototype will be a component itself, just as all other components that a Novulo application is composed of. This means that the style and interface of the component are the same as the rest of the application. No further screenshots or clarification are therefore needed when the working process starts, making the transition of theoretical explanation of the tool to practical daily use a lot less complicated.

An introduction can be assigned to and completed by a certain person and this progress can be monitored using the graphical functions included in the Novulo software. An example of this is provided below, with a graph showing the number of introductions started and completed by the various users.

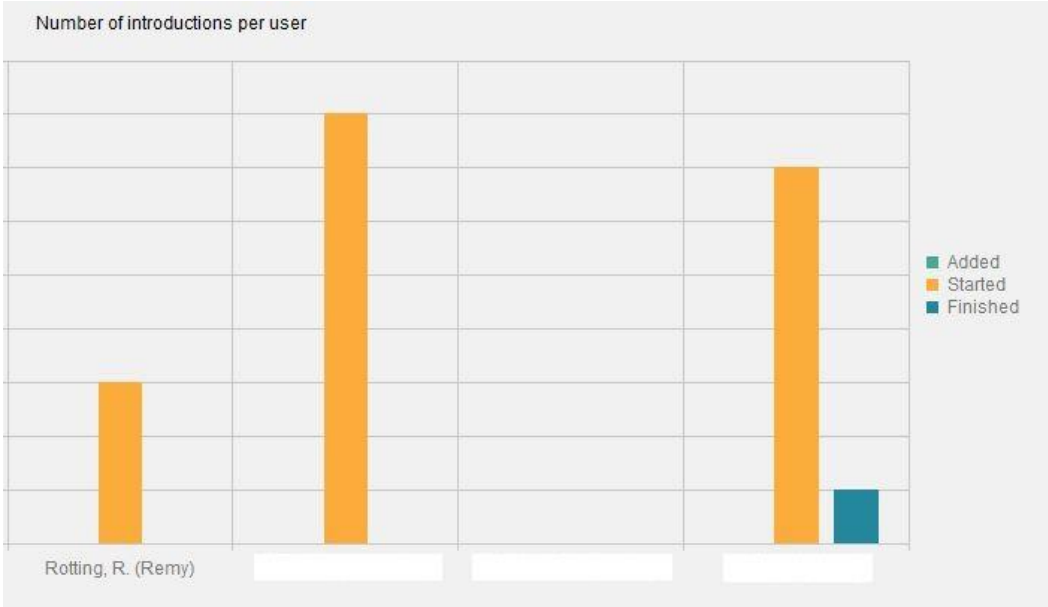


Figure 19: Example graph (# of introductions)

5.3.3 The implementation process

During the workshop, the user will be asked to implement a system that provides some of the most common functions of an Enterprise system, namely the CRM and sales and purchase registration.

To do this, they will follow the implementation path specified in chapter 3 and will be guided on this path by the prototype. This guidance helps them with the implementation activities and provides the necessary knowledge of the implementation process, that is normally provided by the implementation consultants.

The knowledge contained within the prototype tool includes:

1. The implementation process
2. General use instruction
3. Business process instruction
4. Software configuration instruction
5. Data migration instruction

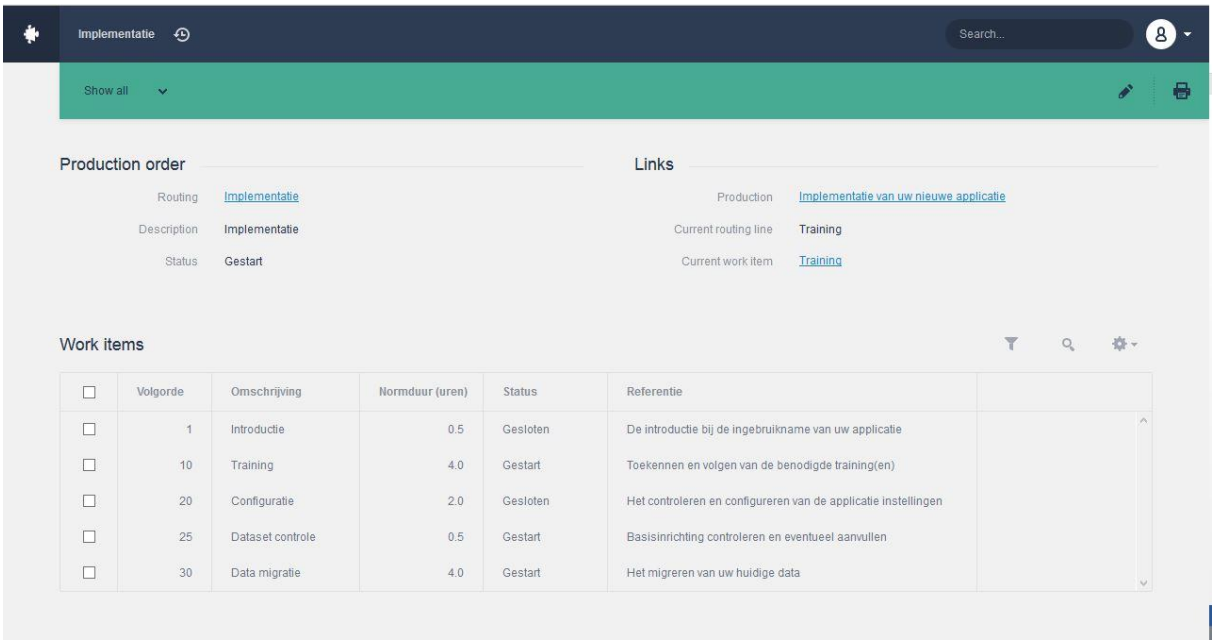


Figure 20: The implementation process within the experiment

Iteration

The knowledge about the process will be provided in the form of a **step-by-step process**, which describes the phases in the implementation. Since the implementation process can be iterative, the steps will have to be taken more than once if change is wanted / needed. Since the various implementing activities and steps can be processed in a short amount of time, iterations pose little problem during the implementation.

This iteration has already been shown in chapter 3, where modifications to the application can influence the amount of time and effort needed during that stage or phase. The phase is completed when the decision is made to stick to the choices made, allowing the next phase to begin.

Implementation process case (running example)

As a running example of an implementation of a web based enterprise system in an SME, the case of a Novulo software implementation at one of their customers is used. A retailer found that his growth could no longer be supported by its current information system, which comprised several Excel sheets and a basic accounting system. The retailer had little knowledge about IT. The choice was made to start with a small application, that could support their CRM, sales- and purchase registration, and finances.

The employee that had been the main user of the accounting system was selected as the main user (champion) and was instructed by a consultant from Novulo in the basic use of the system, and the basics of the business processes, configuration and data conversion.

After this instruction, the main user made the choice to provide the data for the conversion himself and configure the documents in collaboration with the consultant from Novulo. The knowledge gained from working on the documents and the conversion led to an improved understanding of the system, providing the user with valuable knowledge about IT and their own business processes, while cutting consultancy costs.

The choice to start with a small application made sure the business processes remained clear. With the improved understanding developed during the implementation, other processes could be integrated into the application, like tooling concerning marketing. The use of consultants during this project was minimal; with just 12 hours of consulting the project was successfully completed.

6 VALIDATION

The experiment will test the practical value of the model that has been composed by the literature research and the case research done at Novulo. Goal of the experiment will be to test the possibility of implementing the theoretical framework into practical situations. These situations will be tested using the resources that are available at Novulo, namely the Novulo App store and an implementation tool build using the framework.

The experiment will use the factors provided by the model to try and provide prove of further causal relations between the factors that influence the amount of effort and the outcome (the ability to implement a system). The method is deductive, starting with a theory (or in this case, multiple theories combined) and using empirical research to prove the theory.

The research design is described in chapter 5.1, with the general research design explained first, the validity in 5.1.1, the causal relationships of the experiment in chapter 5.1.2, followed by the sample selection and the analysis tools used.

Chapter 5.2 describes the set-up of the experiment, including a timetable and the activities done during the experiment.

The results of the experiment are show in chapter 5.3, with the analysis shown in chapter 5.4. This analysis will show the rate of success during the various implementing activities, thus showing the validity and usefulness of the prototype. The analysis will discuss the results of the questionnaires filled in after each introduction, showing where the tool can be used successfully.

6.1 EXPERIMENT

Seven SME owners have been found that are willing to cooperate in the experimental workshop. During the workshop the contenders will follow a schedule that will guide them through the implementation process of a pre-determined application. The workshop will take approximately 3-4 hours. The schedule below is to be used:

09:00 – 09:15 am Start: Instruction into the experiment using PowerPoint presentation

09:15 – 10:15 am Implementation: General instructions

10:15 – 11:00 am Implementation: Process instructions

11:00 – 11:15 am Short break

11:15 – 12:00 am Implementation: Configuration instructions

12:00 – 12:45 am Implementation: Data migration instructions

12:45 – 13:00 am End: Discuss instructions and overall workshop

The four types of instructions each have two separate introductions. After each introduction, a questionnaire is filled in. When a set of instructions is completed, the user returns to the main process, which will describe the next action for them.

The questionnaires the contenders filled in during the introductions in the workshop have a direct and real time impact on the results, which provides a nice bonus at the end of the workshop: the results are already available right after the experiment and can be viewed in a graph. They can even be viewed during the workshop, to show the impact of the scorings to the implementer, making them more connected to the outcomes.

6.2 DATA COLLECTION

The experiment design follows a pretest / posttest design, where in the pretest the IT knowledge and is examined and in the posttest the same measurement is done after the use of the implementation tool. The experiment will be conducted by doing a **workshop** with owners of SMEs that have no knowledge of Novulo software and will be asked to implement a system that has been specified for them with the tools provided in the prototype.

The goal is to analyze if the implementers can successfully complete the implementation process described by the tool.

6.2.1 Validity

To clarify the experiment and the variables involved in it, the UTOS model by Cronbach (1982) is used. The model describes the Units, Treatments, Observations, and Settings used in the experiment, to provide a framework that can be used for establishing valid causal inference.

- Units: SME owners / lead users
- Treatments: Prototype tool
- Observations: Workshop
- Settings: Workshop with implementation of predetermined ES

Pretest knowledge of Excel and business software

- Question about current IT/IS knowledge

Posttest including:

- Usability
- Satisfaction
- Completion of (implementation) activities

6.2.2 Causal relationships

The experiment will try to provide further proof of the validity of the framework, from a more practical perspective. The causal relationships tested in the experiment are therefore also more practical and aimed at the implementing company and the implementation.

The hypotheses described in chapter 3.3.2 are the main relationships that should be examined. These relations are summed up below:

Table 13: Causal relationships in the experiment

X	-	Y
Tool (production)	increases	Project success
Tool (general)	increases	System usability and satisfaction
Tool (training)	increases	System usability and satisfaction
Tool (configuration)	increases	System usability and satisfaction
Tool (data)	increases	System usability and satisfaction

The causal relationships will be used to explain the hypotheses that were set in the chapters above. These hypotheses have the same variables that need to be tested to be able to validate them. They are listed below, with the corresponding required results.

H_1: It is possible to develop a project planning for the implementation of a web based enterprise system using only a tool provided by the vendor

a) Usability scorings and completion of implementation process set by tool

H_2: It is possible for the implementing company to instruct the other users in their daily processes using only the training included in a tool provided by the vendor

b) Usability scorings, satisfaction, and completion of implementing activities

H_3: It is possible for the implementing company to fully configure the enterprise system using only a tool provided by the vendor

c) Usability scorings and completion of implementing activities

H_4: It is possible for the implementing company to complete the conversion and migration of legacy data using only a tool provided by the vendor

d) Usability scorings and completion of implementing activities

H_5: It is possible for the implementing company to validate the functionality of a web based enterprise system using only a tool provided by the vendor

e) Usability scorings and completion of implementation process set by tool

6.2.3 Sample selection

The sample should be an accurate representation of the target users. The target users of the implementation tool are decision makers and end users of SME's that want to implement an own Enterprise system. These users suffer from the scarcity in resources that is common in SME's and in general have little knowledge of IS and IT. Therefore the choice has been made to select owners of SMEs as the appropriate sample. The participating SMEs have been selected based on personal contacts and business contacts coming from Novulo.

6.3 ANALYSIS

In this chapter the methods and techniques used to analyze the results are described. The chapter starts with the description of the tools and their connection to the integrative model and ends with a critical analysis of the outcomes.

6.3.1 Analysis tools

To analyze the results coming from the workshops, certain tools have to be implemented into the workshop. These tools will be discussed here, starting with the System Usability Scale (or SUS). This scale examines the usability and effectiveness of a system, which is perfect for the validation of the prototype. The scale was first demonstrated by Kirakowski during a Cambridge conference in 1988 [49].

System Usability Scale

The system usability scale is a "reliable, low-cost usability scale that can be used for global assessments of systems usability." SUS has been made freely available for use in usability assessment, and has been used for a variety of research projects and industrial evaluations.

The system usability scale questions are shown in the appendix. It can be seen that the selected statements cover a variety of aspects of system usability, such as the need for support, training, and complexity, and thus have a high level of face validity for measuring usability of a system.

Using SUS

The SUS is generally used after the respondent has had an opportunity to use the system being evaluated, but before any debriefing or discussion takes place. Respondents should be asked to record their immediate response to each item, rather than thinking about items for a long time. All items should be checked. If a respondent feels that they cannot respond to a particular item, they should mark the center point (neutral) of the scale.

Scoring SUS

SUS yields a single number representing a composite measure of the overall usability of the system being studied. To calculate the SUS score, first sum the score contributions from each item. Each items' score contribution will range from 0 to 4. For items 1,3,5,7,and 9 the score contribution is the scale position minus 1. For items 2,4,6,8 and 10, the contribution is 5 minus the scale position. Multiply the sum of the scores by 2.5 to obtain the overall value of SU. SUS scores have a range of 0 to 100.

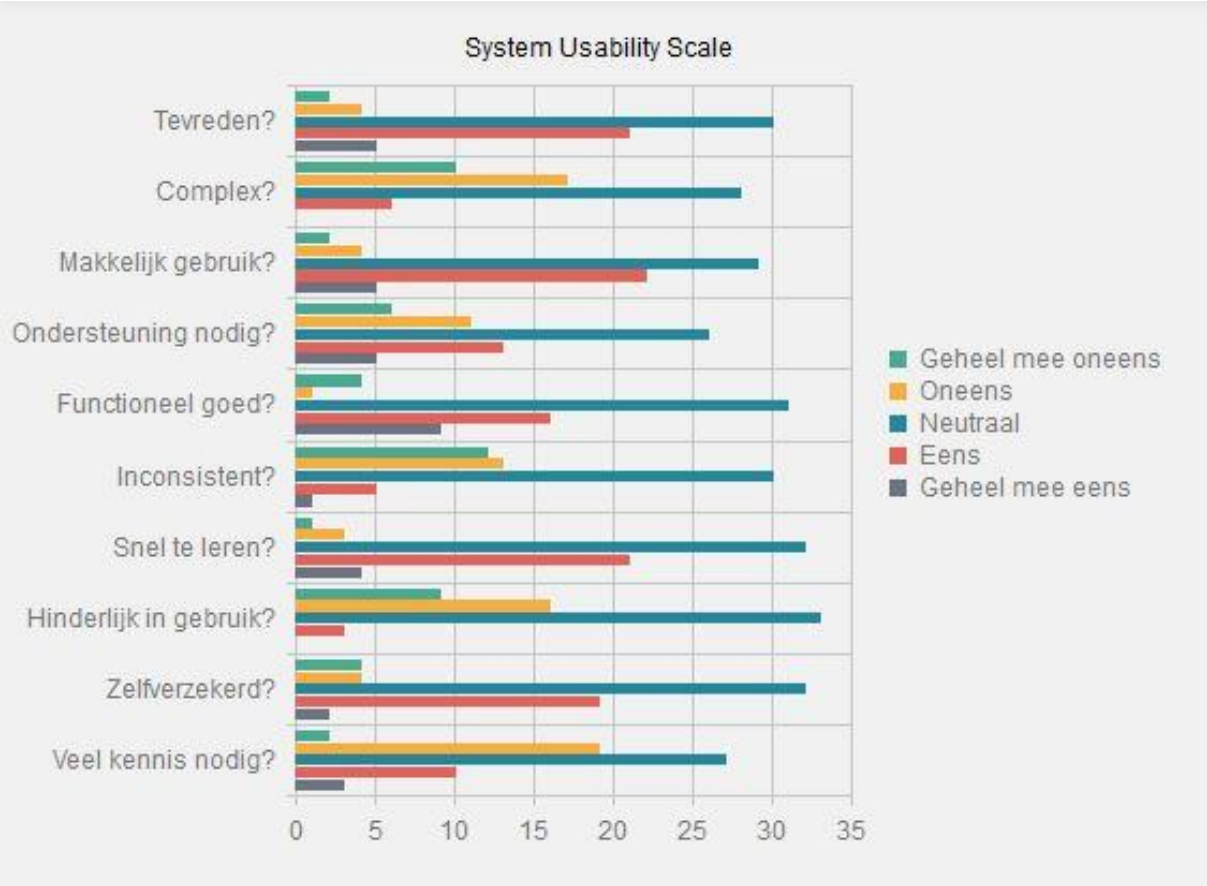


Figure 21: Example SUS results in Novulo (Dutch)

Linking the SUS to the integrative model

The SUS score is the only part of the scale that will be analyzed within this study, the individual items won't be linked to specific questions. The scale is a universal scale, developed specifically to test the usability of a system. This usability itself is the only outcome that is needed to draw conclusions, as can be seen at table 13 (the causal relationships).

Project success and satisfaction

The success of the workshop (the implementation process) is measured in the amount of activities completed by the implementing user and the individual scorings of the question 'I successfully completed this introduction'. This question has items on a 5 point Likert scale ranging from 'Completely disagree' to 'Completely agree', the same as the System Usability Scale scorings. This makes the integration into the survey easier.

The user satisfaction is measured by multiple questions:

- 'This introduction was satisfactory.' and
- 'This introduction was informative.'

These questions are also scored on a Likert scale with the same items as the other items of the survey. The outcome can be calculated by summing the scores and dividing them by the number of filled in questionnaires. This way the mean of the scores is calculated. The score is then multiplied by 100 to find a representative number.

The System Usability Scale, project success and user satisfaction can now all be measured, meaning all necessary information can be gathered from the workshop to receive the results needed to validate the models and the causal relationships linking them.

6.3.2 Outcomes

The answers given to the questionnaires are described in detail below. The SUS scorings provide a general insight into the usefulness and the validity of the four topics discussed in the model (general – training – configuration – migration) and the overall implementation success. The outcomes of the model are described in the table below, where each topic is discussed separately to provide further insight into these topics. The results are compared to the overall implementation success and this overall score can be used to compare the topics to the results.

The success and satisfaction scores are shown in the following order:

1. 'I successfully completed this introduction.' -> Implementation success
2. 'This introduction was satisfactory.' -> Satisfaction
3. 'This introduction was informative.' -> Effectiveness

Overall implementation success

The overall implementation process satisfaction and SUS score can be found by adding all the individual scorings of the 8 introductions together (2 per topic). These can be used to draw an overall conclusion about the usability and usefulness of the prototype. The score can be seen as an affirmation of the usefulness of the tool. The score of 66.40 of 100 is above average and is seen as acceptable. The success (completion in %), satisfaction, and effectiveness score are also positive, showing scorings of 80 and 75 percent.

Table 14: Overall workshop survey scores

Overall (entire process)	Score
System Usability Score	66.40/100
Success (completed) Satisfaction Effectiveness	1. 80% 2. 75% 3. 80%

General

The two general introductions used in the experiment were about the use of the user interface, mutating records within the enterprise system, searching and filtering on one side, and the use of accounts and rights profiles on the other. The introductions were the first real use of the application that the implementing company has, so the basics are described in these introductions.

The scorings in this part showed the difference in the knowledge requirement in different parts of the application; the introduction in the general use was scored highly, but the use and configuration of accounts, and especially rights profiles, were scored significantly lower. These scores together still show a positive outcome, with a score of 63.93.

Table 15: General introductions survey scores

General introductions	Score	Mean (overall score)
System Usability Score	63.93/100	66.40/100

Training

The two training introductions used in the workshop discussed the two main business processes included in the enterprise system: the CRM and sales and purchase registration. The goal of the training introductions was to acquaint the implementing user to the functionality that is included in the system and accommodates their daily processes.

The training instruction showing the business process within the application had the highest usability scorings. They were found to be helpful by all participants and showed that the processes within Novulo can be explained by a tool in such a way that the implementing company can complete the activities linked to these processes successfully. The ability to transfer this type of knowledge was found to be very helpful. This also provided another option for the tool: as a learning tool.

Table 16: Training introductions survey scores

Training introductions	Scores	Mean (overall score)
System Usability Score	75.00/100	66.40/100

Configuration

The two configuration introductions discussed in the experiment showcased the settings that are required to be able to fully use the functionality of the business processes above. The introductions described how to add basic settings supplied by Novulo and how to validate and edit these settings on your own. The settings discussed in the introductions were the general settings and the sales settings.

The configuration of the application requires a certain level of knowledge that is hard to acquire during the first introduction with the software. All participants found completing the configuration the hardest part of the implementation. The validation and set up of the settings was found to be complex and participants could not link the settings back to their context with the amount of knowledge they had gained of the business processes.

The difficulty of the configuration is reflected in the score: the score is 56.79, which is barely above the average of 50. This still shows that the system (tool) is useful and usable, but not as effective as it is in the other introductions. Thankfully this knowledge is not always required during an implementation as market standards can be included by Novulo.

Table 17: Configuration introductions survey scores

Configuration introductions	Scores	Mean (overall score)
System Usability Score	56.79/100	66.40/100

Migration

The first introduction about the data migration described the concepts and options available during a data migration, the risks and benefits of data migration. The second introduction discussed the migration of customer data (organizations, persons, addresses, and bank accounts) to the enterprise system and had an assignment where the implementing user had to actually import this customer data using one of the Excel sheets Novulo provides.

The migration is usually a painstaking process requiring a significant amount of time. The workshop showed that the amount of effort needed to convert and migrate your own data does not have to be that high. With the support of the tool and the Excel sheets provided by Novulo, the participants were able to successfully import some of their customers into the system. They responded positively on the ease of use and comprehensibility of the sheets, showing that the migration of data can be done by the implementing company. The findings are also reflected in the scoring of this part of the introduction, with a score of 64.64.

Table 18: Data migration introductions survey scores

Migration introductions	Scores	Mean (overall score)
System Usability Score	64.64/100	66.40/100

The above results will have to be translated back to the causal relationships and the hypotheses behind them. This is done in the next part of this chapter: the critical analysis.

6.3.3 Critical analysis

Critical analysis is needed to further discuss the usability and effectiveness of the models that were designed in this thesis. The three created models will be discussed here, in their order of appearance in this study. The outcomes of the workshop will be analyzed first, these will be connected to the corresponding models afterwards.

The outcomes of the workshop were generally positive. Most of the topics that were discussed during the workshop have been completed successfully by the implementing company, showing that implementation on your own is a valid option, if the system is easy to use and has little complexity.

Table 19: Causal relationship in the model, including outcomes

X	-	Y	Outcome
Tool (production)	increases	Project success	Positive
Tool (general)	increases	System usability and satisfaction	Positive
Tool (training)	increases	System usability and satisfaction	Positive
Tool (configuration)	increases	System usability and satisfaction	Negative
Tool (data)	increases	System usability and satisfaction	Positive

A further explanation of the outcomes is given below, linking them back to their original hypotheses. The model described at the end of the literature research (chapter 2) can also be linked back to these results.

H_1: It is possible to develop a project planning for the implementation of a web based enterprise system using only a tool provided by the vendor

Positive: The hypotheses has been tested using the overall SUS scores and the satisfaction and training effectiveness results of the survey. The outcomes of the workshop show that successfully completing the implementation process of your own web based enterprise system is possible using the tool provided in the workshop. The structure of the tool that was made according to the literature research proved to be able to guide the participants through the process, which was the main focus of this hypotheses.

H_2: It is possible for the implementing company to instruct the other users in their daily processes using only the training included in a tool provided by the vendor

Positive: The tool proved to be a able to instruct the participants in the registration of the core business processes included in the workshop (CRM, sales- and purchase registration). With the instructions provided by the tool, participants could complete all processes themselves, therefore supporting this hypotheses.

H_3: It is possible for the implementing company to fully configure the enterprise system using only a tool provided by the vendor

Negative: The configuration of the application proved to be too complicated for a first time user to fully understand and complete. The amount of IT and system knowledge required to add or edit settings is too high. However, with the availability of datasets including the best practices of the market, the configuration of the application is not a required part of the implementation process.

H_4: It is possible for the implementing company to complete the conversion and migration of legacy data using only a tool provided by the vendor

Positive: The data migration included in the workshop could successfully be completed using the tool and the material provided. This implies that the migration of data can also be done by the implementing company. However the data migration in the workshop only included the migration organizations, persons, addresses, and bank accounts. This is the core data usually encountered during data conversion and migration. In a SAAS implementation the amount of data that needs to migrated is usually higher and more complex, which could prove to be too difficult for the implementing company to complete on their own.

6.3.3.1 The implementation process

The process of a web based SAAS implementation has been designed based on the model by Markus and Tanis. The implementing activities remaining in a SAAS implementation have been identified, implemented in the integrative model, and tested in the workshop. This SAAS implementation process and the literature research on general ES implementation are the foundation of the model that led to the tool. With the success of the workshop, especially in the amount of activities successfully completed and the satisfaction, the implementation process as depicted in this research has been proven to be valid. Further research is needed however to see if the process discusses all activities during an implementation, not just the activities that are required to be able to do a satisfactory web based ES implementation.

6.3.3.2 The integrative model

The integrative model that combines the implementation process and the outcome of the literature research have been used as the backbone of the prototype. The framework set in the model formed the prototype, and by testing the prototype this form of the tool could be analyzed.

6.3.3.3 The prototype

The prototype has been extensively tested during the workshops. The tool showed that it was capable in supporting the implementing company during the implementing activities in the general, training, and conversion parts of the implementation. The configuration of the application still has elements that are too complex to fully complete.

7 CONCLUSION

The goals of this study were threefold:

1. Create a model web based implementation process in SMEs
2. Combine the implementation model with literature research to create an integrative model with all the factors influencing the ability to implement the enterprise system
3. Create a prototype to validate the above models

The implementation model has been created by combining the research by Markus and Tanis and the results of the interviews held with consultants at Novulo. Using this method ensured that the implementation process by Markus and Tanis, which was developed for on-site implementations, was still viable for web based implementations. The research found this was the case, but many activities described were no longer relevant in a web based environment.

By doing research on the literature available on implementations, an integrative model of the factors influencing the ability of the implementing company to implement their own system has been formed. This model has been applied to the context of an SME that wants to implement a web based Enterprise system that is supplied using a SAAS model.

A workshop has been conducted with a number of SME owners using an experimental model constructed according to the literature. This workshop has been tested and the results have been discussed in chapter 6. The prototype used in the workshop is based on the model described in the literature review and will test the possibility of implementation done entirely by the implementing organization. This means that all implementing activities named by Markus and Tanis, if relevant for the implementation, will have to be supported by the prototype tool. The design process followed the iteration steps as provided by Peffers.

The conclusions of the workshop have been split into the various implementing activities, providing the possibility to analyze each of the four parts of the implementation (general – training – configuration – migration). This division showed the effectiveness and usability of each part. The results showed that implementing a web based enterprise system on your own is possible, when no configuration is needed besides the (industry) standard provided by the supplier. The usability of the general, training, and migration instructions included in the tool were all found to be positive in the surveys held during the workshop, the instructions regarding the configuration were found to be too complex by most attendants.

The conclusions to the research questions are described in the next chapter, showing the answers to each sub question.

7.1 CONCLUDING STATEMENTS

The concluding statements below describe the answers to the sub questions in detail. All the answers combined will provide the answer to the main question of this research:

What factors (methods and techniques) enable the business user to design (based on existing functional components) and implement their own enterprise systems?

This main question will be answered last, by combining the sub questions.

7.1.1 Sub questions

The various sub questions will be discussed first, before the final conclusion will be drawn. The sub questions each represent a part of this study (literature and/or experimental research).

7.1.1.1 What are web based enterprise systems (ES)?

The thesis follows the definition portrayed by several ES and ERP studies, defining an enterprise system as: "*a comprehensive, packaged software solution seeking to integrate the complete range of a business's processes and functions in order to present a holistic view of the business from a single information and IT architecture*" [20].

In this definition, an Enterprise System could include enterprise resource planning (ERP), customer relationship management (CRM), supply chain management (SCM), and e-procurement systems [21]. The terms Enterprise system and ERP system are interchangeably in the context of this study.

With the Software as a Service (SAAS) model a customer contracts the use of an enterprise system, such as ERP or CRM, hosted by a third party, rather than buying a software license and installing the application on its own machines [28]. SAAS has many advantages for the client as well as for the vendor. The most recalled advantages of SAAS for clients are the lack of initial investment, faster implementation, no installation, no hardware issues for the client, and no maintenance processes [29].

7.1.1.2 What is the reference implementation process of a web based ES in (Dutch) SMEs?

In summary, the resources such as time, finance, and expertise, that are necessary for planning, represent the most critical difficulties in small businesses. Inadequate resources spent on the implementation increase the risk of the failure of the ES implementation.

Suppliers have begun to offer information systems according to service based business models, such as Software as a Service and Cloud Computing. These service based business models challenge the conventional payment models like the on premise installations and design of standard application packages, and have implications for both users and suppliers of these systems.

The difference between the implementation of on premise ES, which are regarded as traditional implementations where the system is hosted on-site and the company purchases and "owns" software and software licenses, and SAAS on the other hand, is that it is deployed in a model that allows for the provision and use of an Enterprise system by a vendor or provider via the Internet. In this case, software applications are hosted and provided by a vendor on a subscription or lease model over a network, which typically is the Internet.

In an (Enterprise) system implementation are traditionally three major parties involved: the organization implementing the system (the implementer), the organization that developed the system (vendor), and an organization aiding the implementation (the consultant). Each of these three parties contribute in different ways to the project.

The **implementer** has the detailed knowledge of its own particular business processes, organizational context, and competitive situation, which is essential for successful implementation. The **vendors** provide the implementer with software and offer training programs in connection with their products. The **consultants** are brought into ES implementation projects to provide additional skills, knowledge, or simply manpower that is not available at the implementer or the vendor, or is too expensive if procured from the vendor.

The model used in this study is the model by Markus and Tanis [19]. The model consists of four phases, characterized by key players, typical activities, characteristic problems, appropriate performance metrics, and range of possible outcomes. The phases are: project chartering, the project, shakedown, and the onward and upward phase.

- Project chartering includes the activities before the official start of the project. These include the organizational decision about an investment in a new IS solution, mapping of existing business processes, analysis of potential benefits and limitations, specification of functionality needed, and system selection.
- The project phase encompasses all activities between the system selection and "going-live". It comprises activities such as project team building, business process modeling and reengineering, system customization and configuration, end user training, data conversion, testing and debugging, and rollout.
- Shakedown is defined as the period between "going-live" and the time when operations get into routine use. During this phase the system performance is tuned, bugs are fixed, and additional training is conducted if needed. The end users are getting familiar with the system and operations are becoming normal.
- The onward and upward phase is defined as the period since normal operations to when the system is replaced by an upgraded version or a different system. Typical activities involved are additional user skill building, continuous business improvement, and benefits and success assessment.

7.1.1.3 How can you define the 'ideal' implementation process, which can be built into a tool?

The model of the implementation process by Markus and Tanis will be used as the starting point for the process. The project phase of this model describes the implementing activities that are needed. In a web based implementation certain activities that are needed have already been pre-installed by the vendor. This makes implementing a web-based system a lot easier and faster than the classical on site implementation. To find the activities that are still relevant in a web based implementation, interviews have been held with the consultants at Novulo, asking them what activities were still relevant in a SAAS environment.

These activities will be the activities that have to be taken care of during the implementation of a web based system and being able to complete them will define the ability of the implementing company to implement its own system. The other activities have no further use, since they have are done by the software supplier or have become irrelevant.

- Development of a detailed project plan
- Current and/or future business process modeling and reengineering, if any
- Integration of software bolt-ons and/or legacy systems, if any
- Software configuration
- Data cleanup and conversion
- Executive and end-user training

For each of the implementation phases, the inputs, outputs, activities, and techniques have been identified. The process described the entire implementation, including the possibility of new iteration cycles when changes are desired by the implementing company.

7.1.1.4 What factors influence the amount of effort needed to successfully implement a web based ES in an SME?

The **business architecture** (or organizational context) includes the companies' technical, organizational, and environmental characteristics. These characteristics explain the knowledge, resources and restraints within the implementing company. The characteristics can't be altered by the tool, but most definitely have an impact on the ability to implement an enterprise system. In this research, the SME context is used. By specifying the context, the general characteristics of this context can be used, minimizing the effect on the validity of the research results. Further research should be done to find the influence of these characteristics on the implementing ability of the company however.

The **technical architecture** of the implemented system has a large impact on the amount of activities that are needed to implement the system. An on-site implementation of an enterprise system has too many activities, some can only be done by the vendor, making implementation by the implementing company impossible. Implementing a web based

(SAAS) enterprise system eliminates many of the activities, creating the possibility to implement your own system.

The **system architecture** has an impact on the amount of training that is required to work with the enterprise system. When the system is easy to use and the processes fit with the current business processes, the amount of time to transfer to the new system is decreased. The use of effective support tools during and after implementation can further reduce the need of training.

The factors of the technical and the system architecture influence the secondary factors: the number of implementing activities and the amount of training required. These secondary factors can be influenced by the tool.

The **implementing activities** remaining in a SAAS context have been discussed before. The **training requirement** is determined by the ease of use of the software, the amount of BPR required and knowledge within the company. Since the (starting) knowledge within the company cannot be influenced by the vendor, the other two will remain and have been included in the model. The instructions used in the training are to be included in the tool and should provide the support needed to complete the implementing activities and (after the implementation has been completed) start the daily work processes.

With the ease of use of the system and the fewer remaining activities, it could be possible to implement a system without the use of consultants. The **tool** must ensure that the abilities of the implementing company are sufficient to successfully implement their own system. These factors combined influence the possibility of successful implementation done by the implementing company.

7.1.1.5 How can you design this implementation tool as intuitive as possible?

The design of the prototype should follow the same rules as the model dictates for the Enterprise system itself; it should be easy to use, require no business process reengineering, and have sufficient support within itself. These are the main reasons the prototype has been kept simple in its appearance, there are few options available at first, making it less confusing when the implementing person logs into its application for the first time.

The functionality needed to provide the support the implementing company needs according to the model, is the core of the prototype. If the prototype doesn't support all factors, it is likely the implementation would fail. The hypotheses each cover a different implementing activity, making identifying the factor(s) causing the failure easier.

The implementation process can be divided into several functional parts. These parts are the various implementation activities grouped into four subjects that are held together by the implementation process itself.

These subjects are described in detail below:

- **Implementation process:** The process that encompasses the entire implementation, from start to finish. Since knowledge about this process is lacking in an SME, the tool should guide the implementing company through this process, with clear directions and milestones. The implementation process will be one of the first overviews to be displayed, showing the implementer the steps to be taken and the time that is required for them in a linear fashion.
- **General:** Instructions in the general use of the application. These are application specific, since every software system has a different User Interface and navigation. Subjects of the integrative model included in this segment are the general system design and use of the application, providing the basic knowledge that is essential for the use of the system.
- **Training:** This subject is based on the training activities that are required during an implementation project. It discusses the various processes a user has to do in their daily routines, like customer, sales, or purchase registration. This part of the implementation tool can also be used for the regular training of the users, adding further value to the tool.
- **Configuration:** Configuration is everything that is related to the set-up of the system. Every that can be configured has to be set up in such a way that the implementing company can work optimally with it. Configuring these settings and checking them usually requires knowledge about the enterprise system.
- **Data:** The data subject handles the conversion and migration of the legacy data, if any. The conversion and migration of the data can be done in several ways, depending on the type and brand of legacy system. Options available are the automatic conversion using a toolset (if available), a manual conversion, or no conversion. This choice can be made on several levels, depending on the desired amount of data that needs to be converted.

The knowledge about the process will be provided in the form of a step-by-step process, which describes the phases in the implementation. Since the implementation process can be iterative, the steps will have to be taken more than once if change is wanted / needed.

Since the various implementing activities and steps can be processed in a short amount of time, iterations pose little problems during the implementation.

An interesting way to keep the employees of the implementing company interested in following the prototype is gamification. Gamification is the use of gaming concepts during a non-gaming process [47], to increase immersion and motivation to complete the process. The gamification concept can be seen frequently nowadays, in many different kinds of context. Hamari et al. [48] found in their review that gamification has a positive effect on motivation and engagement in learning tasks, as well as enjoyment over them.

7.2 LIMITATIONS

This study has had multiple factors that are uncontrollable. These factors can have an influence on the results, lowering the external validity. The limitations are described below:

- The chosen market of Dutch SMEs may not be representative for the rest of the world.
- Only SMEs are analyzed, larger companies will most likely have very different needs and different complexities, that aren't as easy to support with a tool.
- The analyzed SMEs have not been selected randomly, lowering the generalizability of the research results.
- The knowledge of the implementer has a large impact on the success of the implementation. Determining this beforehand was not possible due to time constraints.
- Only a small part of the functionality of Enterprise systems has been researched (CRM and sales and purchase registration), other functions could provide different results.
- The Novulo App store and the implementation model used in the experiment are resources that are only available at Novulo. This makes these tools organization specific and hard to generalize. The framework and the model that is behind them is the main focus of this study and is generalizable.

7.3 RECOMMENDATIONS

The research included in this study has shown that implementing an Enterprise system with the current technology (Novulo) and architecture (SAAS) can be done by the implementing company itself, providing it has some IT knowledge, the Enterprise system provides sufficient guidance in the implementation process and its various activities, and little to no configuration needs to be done by the implementing company.

The usability scorings concerning the training instructions also provide a starting point for future training modules with Novulo software, since these scorings show the usefulness of these instructions as a training tool.

With the capabilities Novulo provides with their software, the tool can be further improved to increase its usability and effectiveness.

7.4 SUGGESTIONS FOR FUTURE RESEARCH

The integrative model that has been composed by combining the factors in the literature research has been tested in this study, but more tests are necessary to provide further prove. Furthermore the model itself could be incomplete and missing some vital

components. Testing the validity of the model in a larger experiment / environment could provide more insight into this.

Another important part of this research, the tool, should receive the same further empirical testing. The tool is harder to validate however, since the technology and the software can't be copied that easily by other researchers. Case research at the same software supplier (Novulo) could provide the answer in this regard. The tool is a practical view on the implications of the model and is therefore subjective, meaning a different tool can be made with a better connection to the model.

The literature on the impact of the use of the term apps in enterprise software components is non-existent. Focus of this kind of research could be the impact of marketing an ES as an easy to use and easy to implement addition to the organization. Models that test the willingness and user satisfaction of these systems (like TAM/UTAUT/etc.) can be adopted in this kind of research. In this study, the concept of apps in ES is not the main part of the study, but could be seen as a mediating factor during the experiment. The description of apps in a business environment as it is used in this study can be adopted and/or adapted in future research, giving this study further relevance.

REFERENCES

[1] H. Akkermans, P. Bogerd, E. Yucesan and L. van Wassenhove, "The impact of ERP on SCM: Exploratory findings from a European Delphi Study," *European Journal of Operational Research,* vol. 146, no. 2, pp. 284-298, 2003.

[2] C. Keui, "E-business and ERP: Transforming the enterprise.," *The International Journal of Quality & Reliability Management,* vol. 19, no. 4, pp. 486-487, 2002.

[3] A. Pereira, "Fiscal policy: Lessons from economic research," *Journal of Economic Literature,* vol. 37, no. 2, pp. 688-689, 1999.

[4] L. Koh and M. Simpson, "Could enterprise resource planning create a competitive advantage for small businesses?," *Benchmarking: An International Journal,* vol. 14, no. 1, pp. 59-76, 2007.

[5] E. J. Umble, R. R. Haft and M. M. Umble, "Enterprise resource planning: Implementation procedures," *European Journal of Operational Research,* vol. 146, p. 241–257, 2003.

[6] T. H. Davenport, "Putting the enterprise in enterprise system," *Harvard business review,* no. 76, pp. 121-131, 1998.

[7] P. Bingi, M. K. Sharma and J. K. Godla, "Critical issues affecting an ERP implementation.," *IS Management,* vol. 3, no. 16, pp. 7-14, 1999.

[8] M. Sumner, "Risk factors in enterprise-wide ERP projects," *Journal of information technology,* vol. 4, no. 15, pp. 317-327, 2000.

[9] R. Malhotra, "Critical decisions for ERP integration: Small business issues," *International Journal of Information Management,* vol. 30, pp. 28-37, 2010.

[10] European Commission, "Commission Recommendation 2003/361/EC," *Official Journal of the European Union,* p. 36, 2003.

[11] B. Snider, G. J. da Silveira and J. Balakrishnan, "ERP implementation at SMEs: Analysis of five Canadian cases," *International Journal of Operations and Production Management,* vol. I, no. 29, pp. 4-29, 2009.

[12] P. Achanga, E. Shehab, R. Roy and G. Nelder, "Critical success factors for lean implementation within SMEs," *Journal of Manufacturing Technology Management,* vol. 17, no. 4, pp. 460-471, 2006.

[13] J. Y. Thong, "Resource constraints and information systems implementation in Singaporean small businesses," *The International Journal of Management Science,* no. 29, pp. 143-156, 2001.

[14] S. Mertz, T. Eid and C. Eschinger, Market Trends: Software as a Service in the Enterprise Application Software Markets, Worldwide, Stamford, CT: Gartner, 2007.

[15] Dictionary.com, "application program, in Collins English Dictionary - Complete & Unabridged

10th Edition," HarperCollins Publishers, 2014.

[16] K. Kumar and J. van Hilligersberg, "ERP an Introduction," *Communications of the ACM,* vol. 43, no. 4, pp. 23-26, 2000.

[17] K. Peffers, "A Design Science research methodology for Information Systems Research," *Journal of Management Information Systems,* vol. 24, no. 3, pp. 45-78, 2007.

[18] A. Hevner, S. March and J. Park, "Design Research in Information Systems Research," *MIS Quarterly,* vol. 28, no. 1, pp. 75-105, 2004.

[19] M. L. Markus, S. Axline, D. Petrie and C. Tanis, "Learning from Adopters' experiences with ERP: Problems encountered and success achieved," *Journal of Information Technology,* no. 15, pp. 245-265, 2000.

[20] H. Klaus and G. G. Gable, "What is ERP?," *Information Systems Frontiers,* vol. 2, no. 2, pp. 141-162, 2000.

[21] S. Shang and P. B. Seddon, "Assessing and Managing the benefits of enterprise systems, The Business managers perspective," *Information Systems Journal,* vol. 12, pp. 271-299, 2002.

[22] M. Xin and N. Levina, "Software-as-a-Service Model: Elaborating Client-side Adoption Factors," in *Proceeding of the 29th International Conference on Information Systems*, Paris, 2008.

[23] O. Zach, *ERP System implementation in Small and Medium-Sized Enterprises,* Kristiansand: University of Agder, 2012.

[24] L. G. Tornatzky and M. Fleischer, Processes of Technological Innovation, Lexington, MA: Lexington Books, 1990.

[25] T. Loh and S. Koh, "Critical elements for a succesful enterprise resource planning implementation in small- and medium-sized enterprises," *International Journal of Production Research,* vol. 42, no. 17, pp. 3433-3455, 2004.

[26] W.-L. Shiau, "Development of measures to assess the ERP adoption of small and medium enterprises," *Journal of Enterprise Information Management,* vol. 22, no. 1-2, pp. 99-118, 2009.

[27] S.-I. Chang, S.-Y. Hung, D. C. Yen and P.-J. Lee, "Critical Factors of ERP Adoption for Small- and Medium- Sized Enterprises: An Empirical Study," *Journal of Global Information Management,* vol. 18, no. 3, pp. 82-106, 2010.

[28] A. Dubey and D. Wagle, "http://www.mckinsey.com/insights/mckinsey_quarterly," May 2007. [Online]. Available: http://www.mckinsey.com/client_service/strategy/people/~/media/mckinsey/dotcom/client_service/high%20tech/pdfs/delivering_software_as_a_service.ashx. [Accessed 05 10 2014].

[29] A. Ojala, "Software Renting in the Era of Cloud Computing," in *5th International Conference on Cloud Computing (CLOUD)*, Honolulu, 2012.

[30] S. Lehmann, "Pricing of Software as a Service - An empirical study in view of the Economics of Information Theory," in *Third International Conference, ICSOB 2012*, Cambridge, 2012.

[31] A. Khajeh-Hosseini, D. Greenwood and I. Sommerville, "Cloud Migration: A Case Study of Migrating an Enterprise IT System to IAAS," in *2010 IEEE 3rd International Conference on Cloud Computing*, Miami, 2010.

[32] V. Goncalves and P. Ballon, "Adding value to the network: Mobile operators' experiments with Software-as-a-Service and Platform-as-a-Service models," *Telematics and Informatics,* vol. 28, no. 1, pp. 12-21, 2011.

[33] J. W. Rittinghouse and J. F. Ransome, Cloud computing: Implementation, Management, and Security, Boca Raton: CRC Press, 2010.

[34] R. B. Cooper and R. W. Zmud, "Information Technology Implementation Research: A Technological Diffusion Approach," *Management Science,* vol. 36, no. 2, pp. 123-139, 1990.

[35] J. M. Esteves and J. A. Pastor, "An ERP Life-cycle-based Research Agenda," in *First International Workshop in Enterprise Management and Resource Planning: Methods, tools and architectures - EMRPS'99*, Venice, 1999.

[36] D.-G. Ko, L. J. Kirsch and W. R. King, "Antecedents of Knowledge Transfer from Consultants to Clients in Enterprise System Implementations," *MIS Quarterly,* vol. 29, no. 1, pp. 59-85, 2005.

[37] F. D. Davis, "Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology," *MIS Quarterly,* vol. 13, no. 3, pp. 319-340, 1989.

[38] M. N. Haines and D. L. Goodhue, "Implementation Partner involvement and Knowledge Transfer in the Context of ERP Implementations," *International Journal of Human-Computer Interaction,* vol. 16, no. 1, pp. 23-28, 2003.

[39] Q. Xu and Q. Ma, "Determinants of ERP implementation knowledge transfer," *Information & Management,* vol. 45, pp. 528-539, 2008.

[40] S. Koh, A. Gunasekaran and J. Cooper, "The Demand For Training And Consultancy Investment In SME-specific ERP Systems Implementation And Operation," *International Journal of Production Economics,* no. 122, p. 241–254, 2009.

[41] T. Anderegg, ERP: AZ implementer's guide for success., Resource Publishing, 2000.

[42] A. Parr and G. Shanks, "A model of ERP project implementation," *Journal of Information Technology,* no. 15, pp. 289-303, 200.

[43] S.-I. Chang, D. C. Yen, S.-M. Huang and P.-Q. Hung, "An ERP system Life Cycle-Wide management and support framework for Small- and Medium-sized Companies," *Communications of the Association for Information Systems,* vol. I, no. 22, pp. 275-294, 2008.

[44] J. W. Ross and M. R. Vitale, "The ERP Revolution: Surviving vs. Thriving," *Information Systems Frontiers,* vol. II, no. 2, pp. 233-241, 2000.

[45] F.-N. Nah, J. L.-S. Lau and J. Kuang, "Critical factors for succesful implementation of enterprise systems," *Business process management journal,* vol. 3, no. 7, pp. 285-296, 2001.

[46] C. Francalanci, "Predicting the implementation effort of ERP projects: empirical evidence on

SAP/R3," *Journal of Information Technology,* vol. 16, pp. 33-48, 2001.

[47] S. Deterding, M. Sicart, L. Nacke, K. O'Hara and D. Dixon, "Gamification: Using game design elements in non-gaming contexts," in *ACM CHI Conference on Human Factors in Computing Systems*, Vancouver, 2011.

[48] J. Hamari, J. Koivisto and H. Sarsa, "Does Gamification Work? A literature review of empirical studies on Gamification," in *47th Hawaii International Conference on System Sciences*, Hawaii, 2014.

[49] J. Kirakowski and M. Corbett, "Measuring user satisfaction," in *Proceedings of the Fourth Conference of the British Computer Society on People and computers*, New York, 1988.

[50] P. Kraemmergaard and J. Rose, "Managerial Competences for ERP Journeys," *Information Systems Frontiers,* vol. 4, no. 2, pp. 199-211, 2002.

[51] A. Bhattacherjee, Social Science Research: Principles, Methods and Practice, Florida, 2012.

[52] Z. Huang, "A Compilation Research of ERP Implementation Critical Success Factors," *Issues in Information Systems,* vol. XI, no. 1, pp. 507-512, 2010.

[53] J. Esteves, "A benefits realisation road-map framework for ERP usage in Small and Medium-sized Enterprises," *Journal of Enterprise Information Management,* vol. 22, no. 1, pp. 25-35, 2009.

[54] G. Juell-Skielse and H. Enquist, "Implications of ERP as Service," in *Re-conceptualizing Enterprise Information Systems*, Berlin Heidelberg, Springer, 2012, pp. 129-151.

APPENDICES

LIST OF CONCEPTS

Concept	Definition
SAAS	Software that is licensed to the user, often via the internet. A monthly license fee is often used.
SME	The category of micro, small and medium-sized enterprises (SMEs) is made up of enterprises which employ fewer than 250 persons and which have an annual turnover not exceeding 50 million euro, and/or an annual balance sheet total not exceeding 43 million euro [10].
Implementing company	The company that is adopting the enterprise system.
Vendor	The company that created, maintains, sells, and deploys the enterprise system.
Effort	Hours needed to complete activities.
System architecture	The architecture of the ES; the type of hardware that is used to host the software.
System design	The design of the Enterprise system.
Implementing activities	Activities that are needed during the implementation phase.
Training requirement	The amount of training hours that is required.
Implementing ability	The ability to implement the system set in amount of activities that can successfully be completed.
Tool	The implementation tool prototype that is to be designed and tested during the experiment.

SME CHARACTERISTICS

Overviews of the different SME characteristics that influence the implementation of an Enterprise system are detailed below. The model is based on a study by Zach [23].

SME characteristic	Description
Organizational characteristics	**Resources** - Modest financial resources - Limited human capital - Limited resources for employees' training
	Ownership, management, and decision making - Owner is the CEO - Time constraints of owner-manager - Few layers of management - Centralized decision-making - Short-term decision-making cycle - Intuitive decision process
	Structure - Simpler, flatter, and less complex structure - Flexible structure and information flows - Single-sited - Organic structure - Limited and unclear division of activities - Low degree of employees specialization
	Culture - Unified culture - Few interest groups - Common corporate mindset - Low resistance to change - Organic and fluid culture - Influenced by owner-managers
	Processes and procedures - Smaller and less complicated processes - More flexible and adaptable processes - Informal rules and procedures - Low degree of standardization and formalization
Environmental characteristics	**Market, customers** - Mostly local and regional market - Normally dependent on a small customer base - Affected by powerful partners in their supply chain
	Uncertainty - High level of environmental uncertainty - Uncertain and unstable environment
Information Systems characteristics	**IS Knowledge** - Limited knowledge of IS - Modest managerial expertise - Limited management attention to IS - Lack of strategic planning of IS
	IT technical expertise - Limited IT/IS in-house technical expertise - Emphasis on packaged applications - Greater reliance on third party
	IS function, IS complexity - IS function in its earlier stages - Subordinated to the accounting function

TABLE 10.2 *(continued)*

Phase Name, Description, and Key Actors	Typical Activities	Common Errors or Problems	Typical Performance Metrics	Possible Outcomes

Project—Configuration, integration, and rollout ("dollars to assets")
• Activities designed to get the system up and running in one or more organizational units
• Project manager, project team members, and a variety of external technical and general management consulting resources, executives (in steering committee capacity), other organizational members (in consultative roles)
	• Development of detailed project plan
	• Ongoing project management
	• Selection and assignment of project team members
	• Training of project team members and acquisition of supportive skills
	• Current and/or future business process modeling and reengineering, if any
	• Execution of change management plan, if any
	• Software configuration
	• Software customization, if any
	• System integration
	• Integration of software bolt-ons and/or legacy systems, if any
	• Data cleanup and conversion
	• Documentation
	• Testing, bug fixing, and rework
	• Executive and end-user training
	• Rollout and startup
		• Staffing project subteams without appropriate cross-functional representation
		• Difficulty acquiring adequate knowledge and skill in software configuration (especially cross-module integration), integration of bolt-ons or legacy systems, and a variety of platform technologies
		• Poor-quality software, documentation, training materials
		• Inadequate knowledge on part of consultants and vendor personnel
		• Configuring software for multiple units on the basis of analyzing only one unit
		• Assuming that end-user training should be funded from operations budgets
		• Configuration errors that require rework if caught
		• Customizations that do not work
		• Failure to manage project scope, schedule, budget
		• Inadequate attention to data cleanup
			• Project performance to schedule, scope, and budget	• Project terminated owing to overruns or intractable technical problems
				• Rollout of some operational enterprise system functionality to one or more organizational units
				  • Functionality, operational performance, and organizational preparation are insufficient to address business needs
				  • Functionality, operational performance, and organizational performance are sufficient to address business needs

IMPLEMENTATION STAGES AND MANAGERIAL COMPETENCES

An addition to the model by Markus and Tannis [19], made by Kraemmergaard & Rose [50]. It discusses the managerial competencies needed at each implementation stage.

Stages	**Managerial activities**	**Competences**
Chartering	- Selection of the software packages according to the overall strategy of the company - Creating an initial project plan with deadlines, budgets and employees - Communication with the organization	- Strategic - Technologic - Project management - Communication
The project	- Making a detailed project plan - Current and/or future business process modeling - Managing the people assigned to the project - Configuration and customization of the system - Communication and training	- Project management - Business process - Leadership - ERP system - Communication - Human Resource
Shakedown	- System performance tuning - Training and/or additional training of users - Managing the people assigned to the project - Communication	- ERP system - Human Resource - Leadership - Communication
Onward and upward	- Continuous business improvement - System upgrading and additional functionalities and/or modules - Spreading the ERP knowledge within the organization - Managing a service department - Alignment of the ERP system within future strategic needs for the company	- Business process - Organizational - ERP system - Technology - Human Resource - Leadership - Strategic

RESPONSES PER IMPLEMENTING ACTIVITY

The responses of the interviewed consultants is detailed below. These responses provided the input needed to clarify the implementation process of a SAAS ES in an SME environment.

- **Execution of change management plans**

  "The change management plans concern the impact of the enterprise system on the processes within the organization and how these impacts can be managed. With the component based buildup of applications, if a company thinks the current business process differ greatly from the processes in the application, they can start with the processes that fit most with the current business processes and adopt the rest later. If this isn't possible, most questions can be answered during the training phase." *Consultant 2:* "With the generic buildup of components (best practices), most companies that already have a legacy system have little problems making the switch to a different system. Any problems found can be addressed during training."

- **Selection and assignment of project team members**

  "Since the implementation in a SAAS environment requires less work, the necessity of selecting and assigning team members to the project has become largely irrelevant. SAAS implementation activities can be completed by one or two persons, if they have some IT knowledge like the use of Excel sheets."

- **Ongoing project management**

  "The project lead time has become a matter of weeks (or even days) instead of months. This makes project management a lot easier. Project management is still a requirement of course, but more as a supportive activity."

- **Software customization, if any**

  "The customization of software is largely done after the implementation of the software, since the software itself is best practice based. The wishes of the implementing company can be granted via the use of additional components, which have also been developed already. If need be, however, a component can be developed specifically for the implementing company. This process is done by the software supplier, making the only activity remaining at the buyer to specify their needs."

- **Testing, bug fixing and rework**

  "The software is 'rented' from the software supplier, placing all the testing, bug fixing and rework activities at the supplier. SAAS software that is non-customized should require no further testing or rework."

- **Training of project team members and acquisition of supportive skills**

  "With the smaller size of the project team (1 or 2 employees usually in SMEs), the large scale training of the project team is no longer an important issue. The executive and end user training is enough in these kinds of projects."

- **System integration**

"The integration of the software into the firm used to be an important step in the implementation of on-site enterprise systems. The framework and architecture that was required to install these systems had a large impact on the implementing company. If the architecture was insufficient, costs would greatly increase. With the move from on-site implementation towards SAAS, which is cloud-based, these architectural needs have mostly disappeared, as the software is hosted by the supplier of the software."

- **Documentation**

  "The documentation during an on-site implementation used to encompass the entirety of the documents that were made during the project. In a SAAS environment less documentation is needed. Training can be provided by (online) documents at the supplier or by calling support, project plans are much smaller, and less communication is required."

- **Rollout and startup**

  "The rollout and startup activities have already been completed by the software supplier. The supplier has to provide the implementing company with a working application. The implementing company has no part in this rollout in a SAAS environment." *Consultant 2:* "The startup of the system is usually decided by the final conversion of the data. It's not an activity in this context, but more of a milestone reached in the implementation process."

THE ACTIVITIES STILL REMAINING IN THE IMPLEMENTATION PROCESS

- **Development of a detailed project plan**

  "The project plan has devolved from a large number of steps to a few, detailed milestones showing the activities that have to be completed by the implementing company concerning the configuration, migration of legacy data and training of employees. These activities do not require an extensive development of a detailed project plan, although a project plan is still helpful in ensuring that work is completed on schedule."

- **Current and/or future business process modeling and reengineering, if any**

  "The processes within a business (especially within the SME context), could differ greatly from the processes in the enterprise system. Therefore one of the most important activities during an implementation will always be the alignment of these business processes to the new processes laid unto them by the system. During a web based implementation, this usually boils down to the successful adoption of the system processes, by ensuring user cooperation by explaining the added value of the difference in work process."

- **Integration of software bolt-ons and/or legacy systems, if any**

  "The integration of existing systems is a necessary evil during most implementations. The current systems can usually be replaced by the new application, but have had long implementations themselves making their replacement costly. This means the

bolt-ons will have to be able to communicate with one another. The development of plug-ins can get costly and must be done by the software vendor."

- **Software configuration**

  "The configuration of the software to the specific needs of the company (and the conversion of the old data) is essential during the implementation. If the application isn't configured in the way the company works or expects to work with it, problems are bound to arise."

- **Data cleanup and conversion**

  "The migration of old (legacy) data from to the new system will remain a challenge. As the complexity of the data increases, with more and more (meta-)data that companies may want to migrate, the importance of a correctly done data cleanup and conversion keeps on growing."

- **Executive and end-user training**

  "The training of the (lead) team members of the implementing company is key during the implementation of a system. The consultant won't be around forever, so there should always be a key person or multiple persons at the company that have been properly trained in the use of the system and can answer the questions of the other users."

INTERVIEW QUESTIONS (DUTCH)

Vragen interviews:

Gericht op de implementatiewerkzaamheden

1. Welke werkzaamheden zijn er tijdens een implementatie?	1. 2.
2. Hoe lang duren deze werkzaamheden (schatting)?	1. 2.
3. Welke werkzaamheden hebben het hoogste risico?	1. 2.
4. Welke werkzaamheden kunnen klanten zelf?	1. 2.
5. Welke werkzaamheden kunnen klanten echt niet zelf?	1. 2.
6. Welke personen bij de klant waren betrokken bij de implementatie?	1. 2.
7. Welke personen waren essentieel tijdens de implementatie?	
8. Heeft het modulaire karakter van Novulo invloed op de implementatie? Waarom?	

Gericht op specifieke werkzaamheden volgens Markus & Tanis

1. Making a detailed project plan: In hoeverre is er een project plan gebruikt?	
2. Current and/or future business process modeling: Is er gekeken naar de invloed van het ES op de huidige en toekomstige bedrijfsprocessen?	
3. Managing the people assigned to the project: Hoe werden de betrokken personen aangestuurd en door wie?	
4. Configuration and customization of the system: Wie heeft de inrichting verzorgd van het systeem en waarom is hier voor gekozen?	
5. Communication and training: Wie was hoofdverantwoordelijk voor de communicatie en wie bepaalde welke training nodig was?	

Gericht op de samenwerking met de klant en training

1. Waar heb je de klant het meest in moeten ondersteunen?	
2. Welke onderdelen van de functionaliteit werden als lastig ervaren (zonder uitleg onmogelijk)?	

3. Wat zijn de belangrijkste aandachtspunten bij de samenwerking met een klant?	
4. Hoeveel uur training wordt er gemiddeld gegeven?	
5. Waarover worden tijdens de training de meeste vragen gesteld?	
6. Denk je dat het mogelijk is voor een klant om de gehele implementatie zelfstandig uit te voeren? Waarom?	

Gericht op Software As A Service

1. Wat zijn de voordelen van het aanbieden van ES als (web based) service voor de klant?	
2. Wat zijn de voordelen van het aanbieden van ES als (web based) service voor de leverancier?	
3. Wat zijn de belangrijkste verschillen tussen de implementatie van een on-site ES vergeleken met een web based ES?	
4. Kan het implementatieproces van een web based ES nog gepland worden aan de hand van de modellen die gebruikt worden voor on site ES?	

Gericht op het resultaat

1. Welke factoren hebben het implementatieresultaat het meest beïnvloedt?	
2. Hoe lang heeft de implementatiefase geduurd?	
3. Wat duurde het langst?	

Gericht op een mogelijke tool

1. Kan een implementatietool alle benodigde informatie verschaffen om de implementatie door de klant uit te laten voeren? Waarom?	
2. Welke onderdelen zou een implementatietool af moeten dekken?	
3. Waar liggen de risico's?	

Overige opmerkingen

SYSTEM USABILITY SCALE QUESTIONS

System Usability Scale

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

Implementations in a SAAS environment require fewer activities to complete. Enterprise systems are becoming less complex and easier to use, and general knowledge of IT within an organization is steadily growing.
With these developments in mind, would it be possible for an organization wanting to adopt an enterprise system using the current available technology? Being able to successfully implement your own system cuts consultancy costs, while providing all the benefits of having an enterprise system.

In the thesis this possibility of implementing your own web based enterprise system as an SME has been examined. To this end multiple models have been created and validated in the study:
(1) a model implementation process of SAAS implementation in an SME, (2) an integrative model combining theory from a literature research and gained knowledge from the implementation process, and (3) a prototype tool created using software (Novulo) to validate the models.

To valide the models multiple workshops have been held. In a workshop an owner of an SME was asked to implement a Novulo enterprise system using the tool that had been developed as a prototype. This tool provided the structure, knowledge, and material needed to configure and use the application, understand the business processes included in the application, and migrate legacy data.

The results of the workshops show that the tool supports the implementing activities regarding the training, general introduction, and (simple) data migration. This shows that implementing your own system is indeed already possible. The configuration of the software is still too complex however, and requires support from a consultant to successfully complete, albeit a significantly lower amount.

Keywords: ES implementation – SAAS – Web based implementation – Consultancy effort – SME - Implementation process – Prototype

/ University of Twente
/ Faculty of Management & Governance
/ Department of Industrial Engineering and Business Information Systems (IEBIS)

/ Supervisor:
Dr. M.A. Iacob
/ Second supervisor:
Dr. ir. A.A.M. Spil